APPLIED COURSE
Ph: +91 844-844-0102

Introduction to Digital Logic Design

Digital System:

→ Aimed to provide solutions to real world problems.

Automobile industry is the biggest domain using digital system.

Ex:- Locking of the door. → opened → Alert → which

Automatic Speed Control System → Traffic density

Smart dash board display → "collect" → "send"
Traffic density ⇓ "GPS"

Image processing → Driver alert system, Fuel Indicator
- → Music
- → Sound
- → Rest → parking

Smart energy system etc. → Remote Control "Mobile"

Automatic parking

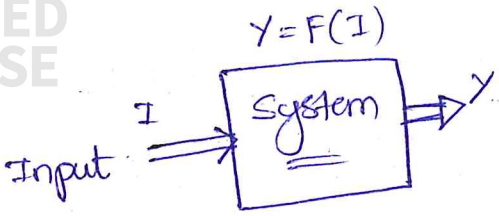
Types of Digital Systems:

process of solving the real world problems.

Example:
1. Headlight warning system → Abnormal warning
2. Door locking alert system → "open" → Alert
3. Railway Signaling → Assigning platform for incoming Train
4. Furnace cooling System
   - → Temparature
     - → Raise
     - → Reduce

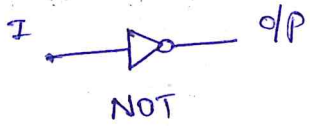
gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102



gate

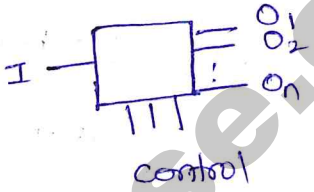
Single I/p	single O/p	NOT
Single I/p	Multi O/p	De-Mux
Multi I/p	single O/p	AND OR
Multi I/p	Multi O/p	



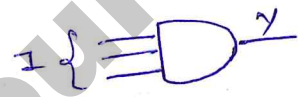
① Combinational $Y = F(I)$

② Sequential $Y = F(I, S)$

State : previous O/p



→ Current output depend on only current input



→ Current output depends on current inputs as well as previous outputs.

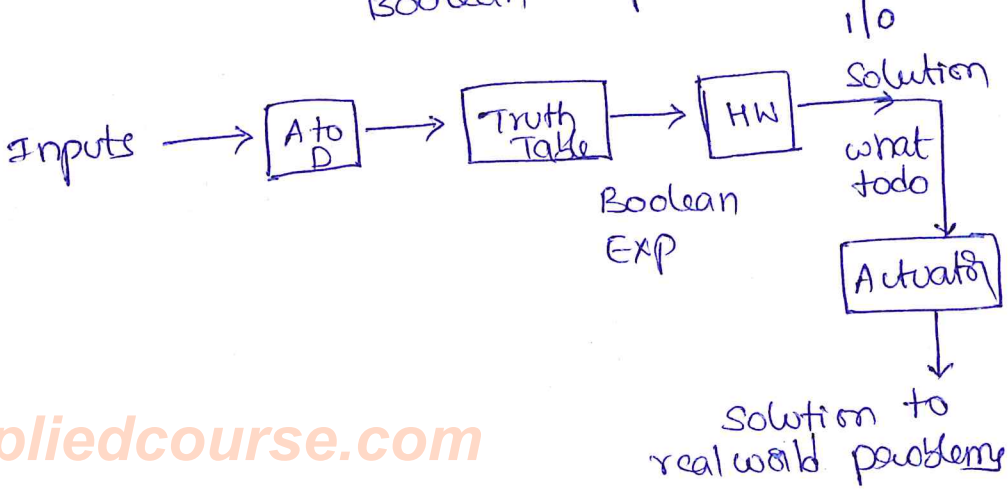
C C S

How real world problems are solved using Digital System

Representation Behavioural capture

⇓

Boolean Expression



gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Input Signals

↓

Truth Table

↓

Boolean Expression

↓

Digital Hardware (Gates, ...)

↓

what is to be done (Switch)

↓

Actuator

Ex: Bank chest opening system

Manager A; cashier B, C

Y : 1 : cash chest opened

0 : not opened

Y=1

A must be

Any one B, C

$Y = AB + AC$

process for Bank chest operation:

In the presence of Manager, one of the cashier has to open the chest.

Bool $AB + AC$

Truth Table:

A B C	Y	
0 0 0	0	Manager is not present ⇒ chest cannot be opened.
0 0 1	0	
0 1 0	0	
0 1 1	0	
1 0 0	0	→ only Manager is present
1 0 1	1	Bank chest is open
1 1 0	1	
1 1 1	1	

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

$Y = A(B+C) = AB + AC$

Ph: +91 844-844-0102

↑ Manager present ↑ Either cashier B or C present

Headlight warning System:

process of solving the Real world problem

→ Identify the system to solve the problem (select i/ps and o/ps)

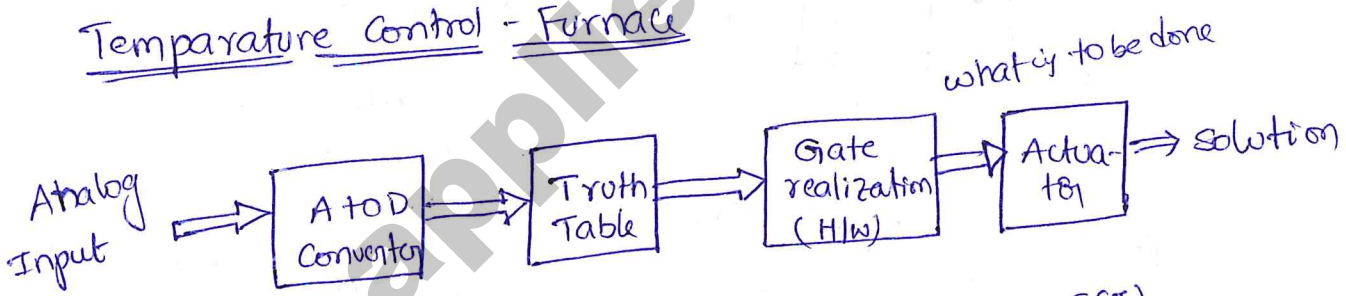
→ Derive the Truth table from system behaviour.

→ Extract boolean expression from truth Table.

→ Solve the Boolean Expression (denote what is to be done)

→ use the appropriate electro-mechanical system to provide solution (Actuator).

Temparature Control - Furnace



Combinational System $Y = F(I)$

Temp > Threshold

Activation of cooling System.

Head light warning System:

Y = 1 warning is given

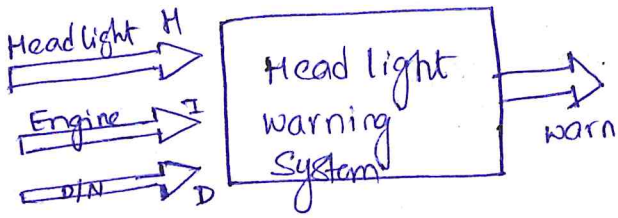
= 0 Normal behaviour

Multi-i/p Single o/p System

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102



Day - Driving - OFF
Night - Driving - ON

when headlight is un-necessarily ON, head light was not 'ON' when needed.

H < 0 OFF, 1 ON
I < 0 OFF, 1 ON
D/N < 0 Day, 1 Night

D=1 & I=1 : Night driving → H must be ON '1'

D=0 & I=1 : Day Driving → H must be OFF '0'

H	I	D	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Truth Table

Normal Y = F(H, I, D)
" : Day Driving } Normal
" warning : Night Driving } Abnormal
unnecessarily Headlight ON
"
Day Driving - Abnormal
Night Driving Normal

Minterm

SOP $Y = \Sigma(3,4,5,6)$

Input Combinations for which output is '1'

POS $Y = \Pi(0,1,2,7)$

Input Combinations for which output is '0'

Maxterm

gate.appliedcourse.com
gatecse@appliedcourse.com

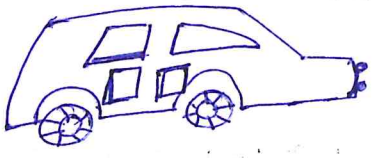
APPLIED COURSE

$Y(H,I,D) = \Sigma(3,4,5,6)$

Ph: +91 844-844-0102

$Y(H,I,D) = \Pi(0,1,2,7)$

Door Locking System:

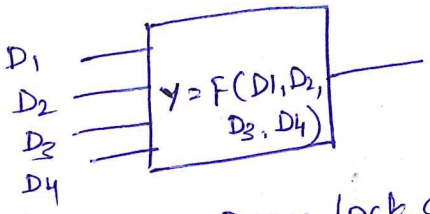


$Y=1$ when any of the door is not locked.

Door Status D_x : 0 closed
1 open

Doors: D_1, D_2, D_3, D_4

$1 \le x \le 4$



Door lock alert

Multi-input Single output System.

D_1	D_2	D_3	D_4	Y
0	0	0	0	0
0	0	0	1	1
0	0	1	0	1
0	0	1	1	1
0	1	0	0	1
0	1	0	1	1
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Alert
Door should be open

Truth Table

gate.appliedcourse.com

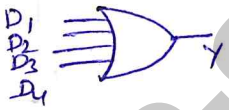
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$Y(D_1, D_2, D_3, D_4) = \Sigma_m(1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15)$

SOP : $D_1'D_2'D_3'D_4 + D_1'D_2'D_3D_4' + D_1'D_2'D_3D_4 + \cdots$
0 0 0 1 0 0 1 0 0 0 1 1

POS : $Y(D_1, D_2, D_3, D_4) = \Pi(0) = (D_1 + D_2 + D_3 + D_4)$

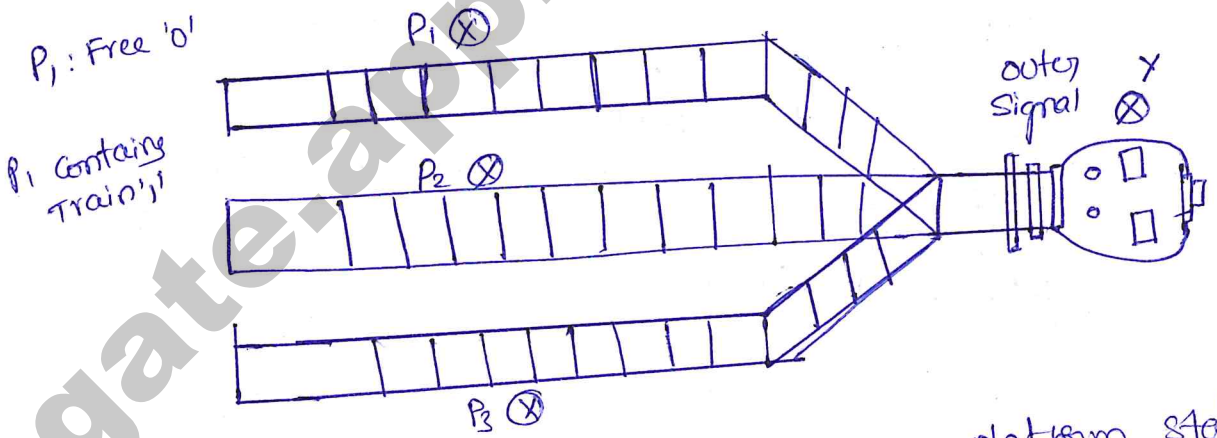


Maxterm:

sum of the input combinations, so that total value of the sum is 0 (output is 0)

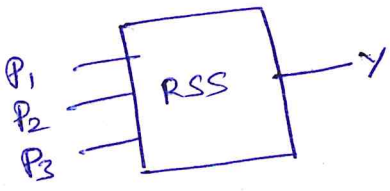
$D_1 D_2 D_3 D_4$
0 1 0 0 0 0
$(0+0+0+0)$
$= 0$

Railway Signaling System:



Multi input single output

platform status P_x:
0 Empty
1: Contains Train



Y : station signal : 0 Not allowed (Red)
: 1 Allowed (Green)

gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Behaviour → Truth Table

Ph: +91 844-844-0102

No waiting at outer

Eight input combinations 0 to 7

Decimal	P_1	P_2	P_3	Y	
0	0	0	0	1	All PF's are Free $Y = G_1$
1	0	0	1	1	P_1 & P_2 are Free $Y = G_1$
2	0	1	0	1	P_1 & P_3 Free
3	0	1	1	1	P_1 Free
4	1	0	0	1	P_2 & P_3 Free
5	1	0	1	1	P_2 Free
6	1	1	0	1	P_3 Free
7	1	1	1	0	-- NO Free PF Y = RED

Truth Table for RSS

MAX term $\Pi(7)$

Minterm $\Sigma(0,1,2,3,4,5,6)$ = GREEN

$$Y(P_1, P_2, P_3) = P_1' P_2' P_3' + \cdots + P_1 P_2 P_3'$$

$$\text{Max term} = 7 \; (P_1' + P_2' + P_3') = (0+0+0) = 0$$

If any one of input is zero output is 1, if all inputs are 1 output is '0'.

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Number Systems:

$O_{r_1} \rightarrow O_{r_2}$

D B

B D

Components: Radix, Digit

'r' 'd'

Constraint: $d < r$

$d > 0$

$r = 2, d = 0, 1$

$r = 3, d = 0, 1, 2$

$r = 7, d = 0, 1, 2 \cdots 6$

Range of digit: 0 to r-1

what happens if radix > 10

Digit encoding using alphabets

Ex:- if $r = 12$, $d = 0$ to $9, a, b$

Common Number Systems

System	Base	Symbols / Digits	Used by Humans	Used in Computers
Decimal	10	0, 1, ... 9	Yes	No
Binary	2	0, 1	No	Yes
octal	8	0, 1 ... 7	No	No
Hexa-decimal	16	0, 1, .. 9 A, B .. F	No	No

IP 192.2.5.133 32bits

(8bits)

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Quantities/Counting..

Decimal	Binary	octal	Hexa-decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

Decimal

0
1
2
:
9
10
11
12
:
19
20
21
:
29

Binary

00
01
10
11
100
101
110
111
1000
1001

If digit $x <$ radix r then digit will be same in radix $r+1, r+2 \ldots$

$(7)_8 = (7)_{10} = (7)_{16}$

octal

0
1
2
:
7
10 - $(8)_{10}$
11 - $(9)_{10}$
:
17 - $(15)_{10}$

Decimal	Binary	octal	Hexa-decimal
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

$r \geq 16$ 0 1 2 ... A .. F Single digit

gatecse@appliedcourse.com
gate.appliedcourse.com

APPLIED COURSE Ph:+91 844-844-0102

Decimal	Binary	Octal	Hexa decimal
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17

Etc..

Inter Conversion:

Integer

① Radix-r to decimal

weighted sum of digits

$\sum N_i d_i \quad N_i = r^i$

iff $d_i \in$ Integer

② Decimal to radix-r

successive division of decimal number by radix-r

stack is used for conversion.

Ex:- $(37)_{10} = (122)_5$

$5 \mid 37$

$5 \mid 7 \quad 2$

$5 \mid 1 \quad 2$

$0 \quad 1$

$\Rightarrow$ | 1 | 2 | 2 | $\Rightarrow (122)_5$

37 two digit number in decimal (10)

⇓

converted into lower radix (5)

⇓

3 digit number

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE Ph: +91 844-844-0102

Higher radix $\xrightarrow{(2)}$ Lower radix $\xrightarrow{(3)}$

Lower radix $\xrightarrow{(2)}$ Higher radix $\xrightarrow{(3)}$

Fraction:-

① Radix r to decimal

Weighted sum of digits

$\sum w_i d_i$; $w_i = r^{-i}$ iff $d_i \in$ fraction

② Decimal to Radix-r

Multiply the fraction with radix r and isolate integer. Queue is used for inter conversion.

Termination condition: Either fraction become zero or significant digits are obtained.

Ex: $(135)_8 =$

$(0.8)_{10} \longrightarrow (\cdot)_5$

$0.8 \times 5 \Rightarrow (\cdot 4)_5$

4.0 stop

$(0.75)_{10} = (\quad)_5$

$= (.3333)_5$

Representational Error

$(0.6)_{10} = (.3)_5$

0.6×5

3.0 ✓ stop

No representational Error

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE

Ph: +91 844-844-0102

InterConversion

$(101.13)_6 = (\quad)_{10}$

$6^2\ 6^1\ 6^0\ 6^{-1}\ 6^{-2}$

$(101)_6 = (6^2*1 + 6^1*0 + 6^0*1)$

$= (36+1)_{10} = (37)_{10}$

$(.13)_6 = (1/6 + 3/6^2) = \frac{6+3}{36} = (\frac{9}{36})_{10}$

$= (1/4) = (0.25)_{10}$

$\sum_i N_i d_i$

$r^i \to d^i \in$ integer

r^{-i}

$(101.13)_6 = (37.25)_{10}$

lower radix → higher

3 digit → 2 digits

No representational Error

Convert Decimal to radix 6 :-

$(37.25)_{10} = (101.13)_6$

6 | 37	1	$(101)_6$
6 | 6	0	
6 | 1	1	
0		

$0.25 * 6$

1.50

$.50 * 6$

3.00

1	3

FIFO

$(37.25)_{10} = (101.13)_6$

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE

Binary to Decimal: & Decimal to Binary

Ph: 91 844-844-0102

$25_{10} = 11001_2 = 31_8 = 19_{16}$

2 | 25
2 | 12 — 1
2 | 6 — 0
2 | 3 — 0
2 | 1 — 1
0 — 1

11001

⇓

1 1 0 0 1
$(2^4\ 2^3\ 2^2\ 2^1\ 2^0)$

$= 1*16 + 1*8 + 1*1$

$= (25)_{10}$

⇓

$(\ )_8$

8 | 25
8 | 3 — 1
0 — 3

$\Rightarrow (31)_8$

Binary to octal (Direct Conversion)

(3) → (1)

Least significant bit towards most significant bit

011|001

$(31)_8$

$(25)_{10}$

⇓

$(\ )_{16}$

$\Rightarrow$

16 | 25
16 | 1 — 9
0 — 1

$(19)_{16}$

Direct Conversion:

$2^4 = 16^1$

$1|1001 \Rightarrow 0001|1001 \Rightarrow (19)_{16}$

Binary to Decimal

→ Multiply each bit by 2^n, where n is the weight of the bit.

→ The weight is the position of the bit, starting from 0 on the right.

→ Add the results.

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Ex 1: $1\ 0\ 1\ 1\ 0\ 1\ 0\ 1$

$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$\Rightarrow 128 + 32 + 16 + 4 + 1 = (181)_{10}$

Ex 2:- $1\ 0\ 1\ 0\ 1\ 1$

$= 2^0 \times 1 + 2^1 \times 1 + 2^3 \times 1 + 2^5 \times 1$

$= 1 + 2 + 8 + 32 = (43)_{10}$

Octal to Decimal:

→ Multiply each bit by 8^n, where n is the weight of the bit.

→ The weight is the position of the bit, starting from 0 on the right.

→ Add the results.

$(724)_8 \Rightarrow 7 \times 8^2 + 2 \times 8^1 + 4 \times 8^0$

$= 448 + 16 + 4 = (468)_{10}$

Hexa decimal to decimal:

$ABC_{16} \rightarrow A \times 16^2 + B \times 16^1 + C \times 16^0$

$\rightarrow 10 \times 256 + 11 \times 16 + 12 \times 1$

$\rightarrow (2748)_{10}$

Octal to Binary:

$(705)_8 = (\ )_2$

$\Downarrow$

$(111000101)_2$

gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Hexadecimal to Binary:

$2^4 = 16^1$

$(10AF)_{16} = (\ )_2$

↓

$(0001\ 0000\ 1010\ 1111)_2$

Decimal to Hexadecimal:

$(1234)_{10} = (\ )_{16}$

16 | 1234		
16 | 77	2	
16 | 4	13	$\Rightarrow (4D2)_{16}$
16 | 0	4	

Octal to Hexadecimal:

use binary as intermediate

Ex:- $1076_8 = ?_{16}$

Octal	1	0	7	6
↓	↓	↓	↓	↓
Binary	001	000	111	110
		2	3	E
↓				
Hexa decimal	23EH			
	0X23E			
	0x23e			

Hexadecimal to octal

Ex:- $1F0C_{16} = (\ )_8$

Hexa	1	F	0	C
↓	0001	1111	0000	1100
B	1	7	4	14
↓				
Octal				

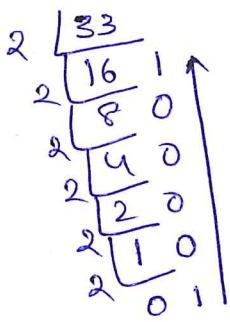
$\Rightarrow (17414)_8$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Ex:

Decimal	Binary	octal	Hexa-decimal
33	100001	41	21
117	1110101	165	75
		703	
			1AF



$\Rightarrow$

1110101

$2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

$64+32+16+5$

96
21
117

1000001

2 1

Example 1: Gate 2004

If 73_x (in base x number system) is equal to $(54)_y$ (in base y number system), the possible values of x and y are

A. 8, 16

B. 10, 12

C. 9, 13

~~D.~~ 8, 11

$(73)_x = (54)_y$

$(7x+3)_{10} = (5y+4)_{10}$

Constraint is $x > 7$ Constraint $y > 5$

$\Rightarrow 7x+3 = 5y+4$

$\Rightarrow 7x - 5y = 1$

$\Rightarrow 7x - 5y - 1 = 0$

Let $x = 8 \Rightarrow 7\times8 - 5y = 1$

$\Rightarrow 56 - 1 = 5y$

$y = 55/5 = 11$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$x = 10$ $\quad$ $x = 9$

$\Rightarrow (7 \times 10) - 1 = 5y$ $\quad$ $7 \times 9 - 1 = 5y$

$\Rightarrow y = \frac{69}{5}$ $\quad$ $5y = 62 \Rightarrow y = \frac{62}{5}$

Gate 2008

Let r denote number system radix. The only value(s) of r that satisfy equation $\sqrt{121}_r = 11_r$ is/are

A. decimal 10

B. decimal 11

C. decimal 10 and 11

D. any value > 2

$\sqrt{121}_r = 11_r$

$\Rightarrow r = 10$ $\quad \sqrt{(121)}_{10} = (11)_{10}$

$\Rightarrow r = 11 \Rightarrow (121)_{11}$

$= (1 \times 11^2 + 2 \times 11^1 + 1)$

$= 121 + 22 + 1$

$= (144)_{10}$

$(11)_{11} = 11 + 1 = (12)_{10}$

If $r = 4$

$\Rightarrow \sqrt{121}_4 = 1 \times 4^2 + 2 \times 4 + 1$

$= 16 + 8 + 1$

$= 25$

$\Rightarrow (11)_4 = 4 + 1 = 5$

For any value $r > 2$, given condition is satisfied.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Gate 2014

The base (or radix) of the number system such that the following equation holds is ———

$312/20 = 13.1$

$\Rightarrow \frac{(312)_r}{(20)_r} = (13.1)_r$ Constraint $r > 3$

$(13.1)_r = (r + 3 + \frac{1}{r})_{10}$

$\Rightarrow (312)_r = (20)_r * (13.1)_r$

$= (20)_r (r + 3 + 1/r)$

$\Rightarrow (3*r^2 + r + 2)_{10} = (2r^2 + 3*2r + 2)_{10}$

$\Rightarrow 3r^2 + r + 2 = 2r^2 + 6r + 2$

$\Rightarrow r^2 - 5r = 0$

$\Rightarrow r = 0$ or 5

Constraint is $r > 3$

$\Rightarrow \boxed{r = 5}$

Gate 2014:

Consider the equation $(123)_5 = (XY)_Y$ with X and Y unknown. The number of possible solutions is 3

$(123)_5 = (XY)_Y$ $\quad X < Y$ or $Y > X$

$\Rightarrow (5^2 + 5*2 + 3)_{10} = (XY + Y)_{10}$ $\quad Y > 8$

$\Rightarrow 25 + 10 + 3 = XY + Y$

$XY = 30$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

	x	y
③ possibilities	1	30
	2	15
	3	10
	5	6
	6	5
	10	3
	15	2
	1	30

$x < y$ & $y > 8$

Answer: 3

Solved problems: $x_3 x_2 x_1 x_0 . y_1 y_2$ format

S1 value : 10.5
S2 value : 4.05
S3 value : 1.005
S4 value : 3.0

which decimal values are not represented by above format.

A. S1, S2
B. S2, S3
C. S1, S4
D. S3, S4

$. y_1 y_2$
$2^{-1}\ 2^{-2}$
$\frac{1}{2^2} = \frac{1}{4} = 0.25$

$\Rightarrow$ $.05 < 0.25$
$.005 < 0.25$

S1 $(10.5)_{10}$

$(1010.10)_2$
Exactly representable

S4 $(3.0)_{10} = (0011.00)_2$
S4 is exactly representable

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Ex: Given $(135)_x + (144)_x = (323)_x$. What is the value of base x?

```
  135
  144
 1 1
-----
  323
```

$(9)_{10} = (13)_r$

$= (13)_6$

$= 6+3 = 9$

$\Rightarrow (323)_6$

$(8)_{10} = (12)_6$

$= 6+2 = 8$

Gray codes, self complement codes, r's complement and r-1's complement:

Gray codes: It is a unit distance, self-reflective, cyclic code. self-reflection is useful for deriving n-bit gray code from (n-1)bit gray code.

1-bit

0
1

2-bit

00
01
11
10

3-bit

000
001
011
010
110
111
101
100

4-bit

0000
0001
0011
0010
0110
0111
0101
0100
1100
1101
1111
1110
1010
1011
1001
1000

Mod-10 Gray Code

0001
0011
0010
0110
0111
1111
1110
1010
1011
1001

gate-appliedcourse.com

Ph: +91 844-844-0102

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844 844 0102

Self Complement code: Let x is the decimal value of binary pattern 'P' then 9's complement of x i.e 9-x is obtained by complementing bit pattern of P.

Ex:- { 2-4-2-1 BCD Code
{ XS-3 (Non-weighted Code) (or) Excess-3

8-4-2-1 BCD	Decimal value	BCD code 2-4-2-1	XS-3 Code.
0000	0	0000	0011
0001	1	0001	0100
0010	2	0011	0101
0011	3		0111
0100	4	0100	1000
0101	5	1011	1001
0110	6	1100	1010
0111	7	1101	1011
1000	8	1110	1100
1001	9	1111	

say 9's Complement of 4 is obtained by subtracting 4 from 9

9-4=5

⇒ 2-4-2-1 code for 4

0100

Complement ↓

1011

2+2+1=5

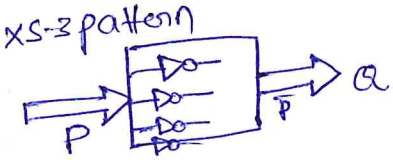
9-4=5

② XS-3 pattern for 4 is

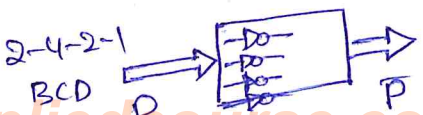
0111

↓ Complement

1000 It is the XS-pattern for 5.



XS-3 pattern of its 9's Complement



9's Complement of P.

By Complementing bits 9's Complement is achieved.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: 191 844-844-0102

Complement and Diminished Complements:

(Radix-r and Radix r-1)

Complement and Diminished complement system is used for to perform subtraction as addition.

Relation: Let n-bit/digit number is 'p' then its r complement is $\boxed{r^n - p}$ and (r-1)'s complement is

$\boxed{r^n - p - 1}$

⇒ For binary number it is 2's Complement and 1's Complement.

⇒ For Decimal number it is 10's Complement and 9's Complement

subtraction of X-Y
- X + 2's complement of (Y) } Binary
- X + 1's complement of (Y) }
- X + 10's complement of (Y) } Decimal
- X + 9's complement of (Y) }

This Complementary/Diminished complement approach may result $\boxed{\text{carry}}$ Then result is +ve (X>Y) X-Y>0

If no carry Then result is -ve (X<Y) X-Y<0

carry presence correction: Ignore carry in 2's/10's complementary arithmetic while diminished complement add 1 to result.

Ex: 5-3

X = 5 101

Y = 3 011

2's Complement of 3

$2^3 - 3 = 5$ (or) 1's Complement(3) +1

gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

2's Complement arithmatic $X-Y$

X + 2's Complement (Y)

```
  101
  101
carry 1
 ----
  010   = (2)10
```

$\Downarrow$

$X > Y$

1's Complement arithmatic $X-Y$

$X = 101$

$Y = 011$

$X-Y$ = X + 1's Complement (Y)

```
  101
 (1)100
  001
  → 1 1
 -----
  010 = 2
```

5-3 = 2

Ex: 2 3-5

$X = 3 = 011$

$Y = 5 = 101$

2's Complement (Y)

$\Rightarrow 2^3 - 5 = 3 = 011$

2's Complement

```
  011
  011
  ---
  110
```

No carry result is -ve

Correct Answer is 2's complement of the result

$(3)_{10} - (5)_{10} = -2$

1's Complement

3 - 011

1's Complement (5)

$\Rightarrow 2^3 - 5 - 1 = 2 = 010$

```
101 ↓ 1's comp  =>  011
010                 010
                   ----
                    101
```

No carry. Result is -ve. Correct Answer is complement of result

$(3) - (5) = -2$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91-814-344-0102

10's Complement & 9's Complement Arithmatic

$(45)_{10} - (34)_{10} = x - y$

$(45)_{10}$ + 10's Complement (34)

⇓

$\Rightarrow \quad 10^2 - 34 = 66$

$$\begin{array}{r} 45 \\ +\ 66 \\ \hline (1)\,11 \end{array}$$

$(11)_{10}$

carry = 1 result is +ve.

9's Complement Arithmatic

$(45)_{10} - (34)_{10}$

⇓

9's Complement (34)

⇓

$10^2 - 34 - 1 = 65$

$$\begin{array}{r} 45 \\ +\ 65 \\ \hline (1)\,10 \end{array}$$

carry = 1

result is +ve and '1' is added to the result. Correct Answer is $= (10)_{10} + (1)_{10} = (11)_{10}$

Ex 2: $(34)_{10} - (45)_{10} = (-11)_{10}$

$$\begin{array}{r} 34 \\ +\ 55 \\ \hline 89 \end{array}$$

No carry. ⇒ Result is Negative.

10's Complement (89) $= 10^2 - 89 = 11$

$\Rightarrow$ (-11)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

9's Complement

$(34)_{10} - (45)_{10} \Rightarrow x = 34,\ y = 45$

45 9's complement is

$10^2 - 45 - 1 = 54$

$$\begin{array}{r} 34 \\ 54 \\ \hline 88 \end{array}$$

No carry result is -ve

9's Complement (88)

$10^2 - 88 - 1 = 100 - 89$

$= 11$

$\boxed{(-11)_{10}}$

	Radix-r	Diminished Radix $r-1$
$X - Y$ $= X + \$(Y)$	case 1: cy=1 Result is +ve Ignore carry case 2: cy=0 Result is -ve	case 1: cy=1 Result is +ve Add 1 to result case 2: Result is -ve. -($r-1$ Complement (result))

Example: $(101101)_2 - (100101)_2$

$= (45)_{10} - (37)_{10}$

$= (8)_{10}$

2's Complement $\quad 10\ 1101$

$$\begin{array}{r} 01\ 1\ 0\ 1\ 1\ 1 \\ \hline 00\ 1000 \end{array}$$

cy is 1 $\Rightarrow$ Result is positive

$(001000)_2 = (8)_{10}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Floating point Representation

Ph: +91 844-844-0102

(1) Floating point number Contains

(i) Integer position

(ii) Fractional portion

(iii) Radix position

2.56

↓

Radix position

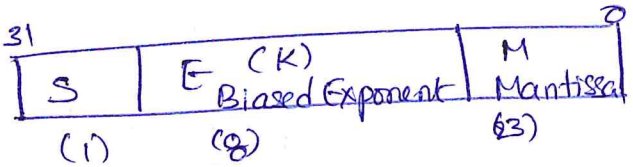
(2) Floating point number requires storing of Radix position.

(3) Most of the systems store Integer and Fractional portion in Mantissa and exponent is stored in Biased Exponent form.

(4) Mantissa is further expressed as normalized fraction.

(5) Biased Exponent is an un-signed integer.

(6) Floating point format



Mantissa

↓

Normalized

Signed Magnitude.

↓

.1xxx Explicit

1.xxx Implicit

$$\text{value}(V) = (-1)^S (1.M)_2 * 2^{E-\text{Bias}}$$

$$\text{Bias} = 2^{k-1}$$

$$= 2^{8-1}$$

$$= 128$$

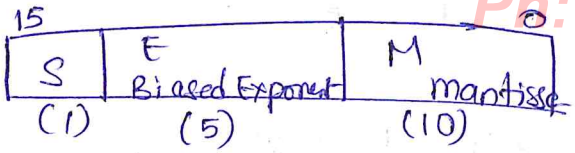
pattern ⟶ value

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Ex:-



Implicit Normalized Mantissa $1.XXXX$

$Bias = 2^{k-1} = 2^{5-1} = 2^4 = 16$ Excess-16 Exponent

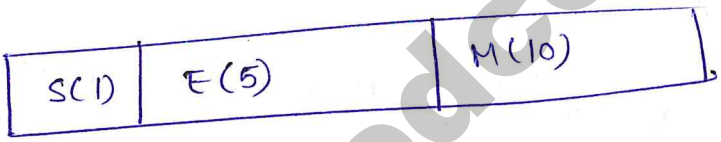
e- True exponent

E → Unsigned number

$E = e + Bias$

$e = E - Bias$

Value $-(7.25)_{10}$ in above format



$S = 1$ $7.25 = 111.01$

Implicit Normalization $= 1.\underbrace{1101}_{M} * 2^2$

$M = 1101000000$

$e = 2 = E - Bias$

$Bias = 2^{5-1} = 2^4 = 16$

$E = 2 + 16 = 18 = 10010$

S	E	M
1	10010	1101000000

C B 4 0

$\Rightarrow (CB40)_{16}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

pattern → value

Largest value

0 11111 1111111111
S E M

$E = 2^5 - 1 = (31)_{10}$

M = 1111111111

$V_{+max} = (-1)^0 (1.1111111111) * 2^{E-16}$

$= (1 + 1 - 2^{-10}) * 2^{31-16}$

$= (2 - 2^{-10}) * 2^{15}$

$= 2^{16}$

Largest value = $2^{16} - 2^5$

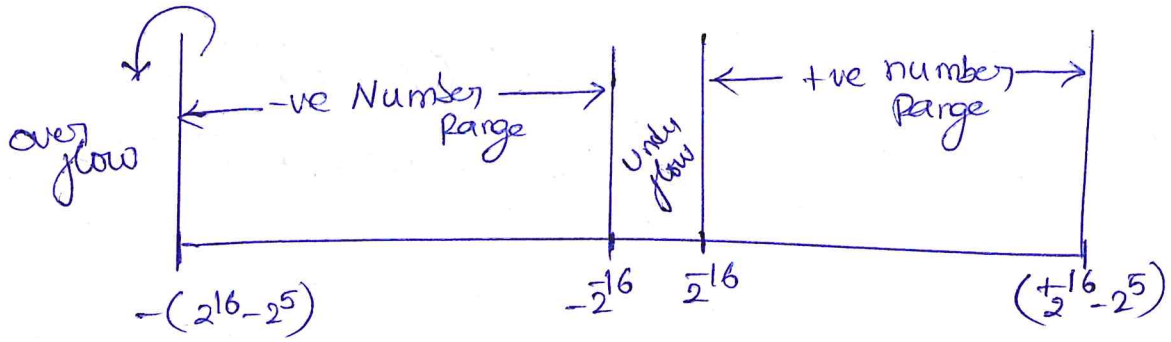
Smallest value:-

S E M
0 00000 0000000000

E=0 M=0

$V_{+min} = (-1)^0 . (1.0000000000)_2 * 2^{0-16}$

$= 2^{-16}$



"zero cannot be represented"

↓

IEEE 754 Resolved

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

IEEE 754 floating point Specification

① This provide for specification of floating point number.
It uses binary radix and biased exponent.

Single precision: S=1, E=8, M=23
32-bits

Double precision: S=1, E=11, M=52
64-bits.

② Floating point number can be stored with
(i) Implicit normalized mantissa $(1.M)_2$
(ii) Fractional mantissa $(.f)_2$

③ Certain mantissa Exponent components denote
(i) Special value $\pm 0, \pm \infty$
(ii) No value NAN (NOT A NUMBER)

④ Value Expression

$$V = (-1)^S (1.M)_2 * 2^{E-127}$$

Bias is 127
$2^{8-1}-1 = 127$

(Implicit Mantissa)

$$V = (-1)^S (.f)_2 \times 2^{-126}$$

(Fractional Mantissa)

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

IEEE 754 Single precession (32-bit)

	S(1)	E(8)	M(23)	value	Expression
①	0/1	00000000 Decimal value E=0	0000.00 M=0	± 0 Special value	—
②	0/1	1111 1111 Decimal value E=255	000...00 M=0	±∞ Special value	—
③	0/1	E≠0 ; E≠255	M=XX...	Implicit value	$(-1)^S(1.M)_2 \times 2^{E-127}$
④	0/1	E=0	M≠0	Fraction	$(-1)^S(.f)_2 \times 2^{-126}$
⑤	0/1	E=255	M≠0	NAN not A Number	

Practice Questions : value → pattern

(i) value $-(12.5)_{10}$ in IEEE 754

(ii) $(0.5)_{10}$

(iii) $(1.5)_{10} * 2^{20}$

$V = -(12.5) \Rightarrow S=1 \Rightarrow (12.5)_{10} = (1100.1)_2$

$= (1.1001)_2 * 2^3$

$\Rightarrow M = \underbrace{100100\cdots0}_{23 \text{ bits}}$

$E-127=3$

$E = 130$ (128+2)

$\Rightarrow (10000010)$

$-(12.5)_{10} = \underbrace{1}_{S}\ \underbrace{1000}_{C}\ \underbrace{0001}_{1}\ \underbrace{0010}_{4}\ \underbrace{0000}_{8}\ \underbrace{0000}_{0}$ (E, M)

$= (C1480000)_{16} = 0XC1480000$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(ii) $(0.5)_{10}$

(iii) $(1.5)_{10} * 2^{20}$

Pattern ⇒ value

Find the value for following 32-bit pattern which denote floating point number in single precision form.

(i) $\underbrace{1\ 10001000\ 0010\ 0010\ 000\cdots0}_{32\text{ bits}}$

$S = 1 \quad E = 10001000 \Rightarrow 128+8 = 136$

$M = 0010\ 0010\ 000\cdots0 = \frac{1}{2^3} + \frac{1}{2^7}$

$(-1)^S (1.M)_2 * 2^{E-127} \Rightarrow$

value $V = (-1)^1 (1.0010\ 0010)_2 * 2^{E-127}$

$= -\left(1 + \frac{1}{2^3} + \frac{1}{2^7}\right) * 2^{136-127}$

$= -\left(1 + \frac{1}{2^3} + \frac{1}{2^7}\right) * 2^9$

$= -(2^9 + 2^6 + 2^2)$

$= -(512 + 64 + 4)$

$= -(580)_{10}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 044 4014-0102

Gate Solved problems on IEEE floating point numbers

① 0 01111101 01010101000...0

S E M

$\Rightarrow S=0 \quad E=125 \quad \frac{1}{2^2}+\frac{1}{2^4}+\frac{1}{2^6}+\frac{1}{2^8}$

$$V = (-1)^0 \left(1+\frac{1}{2^2}+\frac{1}{2^4}+\frac{1}{2^6}+\frac{1}{2^8}\right) * 2^{125-127}$$

$$= \left(\frac{2^8+2^6+2^4+2^2+1}{2^8}\right) * 2^{-2}$$

$$= \frac{341}{2^8} \times \frac{1}{2^2} = \frac{341}{2^{10}} = 0.33307$$

$$= \boxed{3.3307 \times 10^{-1}}$$

② $(1.5)_{10} * 2^{20}$

$(1.1)_2 \times 2^{20} \Rightarrow$ E−127 = 20

E = 147

1.M

M = 1

E = 10010011

$2^7\ 2^6\ 2^5\ 2^4\ 2^3\ 2^2\ 2^1\ 2^0$

S(1) E(8) M(23)

0 | 10010011 | 10000...00000

4 9 C 0

$\boxed{0X\ 49C0\ 0000}$ Hexadecimal

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 9441 441 0102

Fixed point Addition:

2's Complement representation is used for signed numbers

$(8)_{10}$ ← 1000 $-(8)_{10}$

$(3)_{10}$ ← 0011 $+(3)_{10}$

$(11)_{10}$ 1011 ⇒ $-(5)_{10}$

Unsigned Addition.

Signed Addition

Fixed point Multiplication

unsigned 7 × 13

0111 × 1101

signed

7 × −3 = −21

```
  0111
 0000
0111
0111
-----
1011011
```

unsigned 7 × 13

$(91)_{10}$ ⇐ 64+16+8+2+1

⇒ −64+16+8+2+1 = −(37) ✗

Fixed point signed Multiplication : Booth's Algorithm

→ Multiplication of fixed point unsigned numbers cannot be extendable for fixed point signed numbers. as sign-bit distributed in the process.

→ Hence separate algorithm is needed for fixed point signed multiplication. Booth's algorithm is one of them and provide fast response.

→ Booths algorithm uses "Booths Notation" for representing Fixed point signed numbers and "Arithmatic Right shift of partial Sum" used in the algorithm.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

→ Booths notation will reduce number of arithmatic operations while Arithmatic Right Shift of partial sum avoid the saving of partial products and product sign bit.

Booths Notation:

It treat the signed binary number as group of 1's and group of 0's.

→ Any signed number can have atmost one partial 1's group and any number of complete group of 1's.

→ Each Complete group require one addition and one subtraction and partial group needs only subtraction.

→ Booths Notation cover same range as 2's Complement Notation. -2^{n-1} to $2^{n-1}-1$

Example:

SNo	Signed Number Binary pattern	Number of 1's Groups		Number of Arithmatic operations	
		Complete	partial	Additions	Subtractions
1	0111 0111	2	0	2	2
2	0111 1110	1	—	1	1
(3)	1111 1100	—	1	—	1
4	1110 0111	1	1	1	2

Better performance, when there is a less number of arithmetic operations.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(II) Booths Recording or Coding

→ The coefficients of fixed point number are coded into −1, +1 and 0.

→ Beginning position bit of 1's block is encoded as −1 and next to end position bit of 1's block is encoded as +1. Rest partial bits in booths recording pattern are 0's.

→ The recording pattern tells how many and what type operations are needed in for this multiplier with respect to Booths Algorithm.

Ex:

0 1 1 1 0 1 1 1

−1	0	0	−1	+1	0	0	−1
7	6	5	4	3	2	1	0

Bit positions

8-bit recording pattern ↑

This multiplier pattern 0111 0111 require 2-additions 2-subtractions and 14 (0+3+4+7) shift operations with Booths Algorithm.

(0+3+4+7): summation of Non-zero

value of the number: weighted sum of coefficients.

positional weights : 128 64 32 16 8 4 2 1

coding pattern : +1 0 0 −1 +1 0 0 −1

value of Number : (+1)*128 + (−1)*16 + (+1)8 + (−1)*1

= 128 − 16 + 8 − 1 = $(119)_{10}$

gateappliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Ph: +91 844-844-0102

Boothe Algorithm Example:

Let Multiplier Q: $Q_3 Q_2 Q_1 Q_0$

Multiplicant M: $M_3 M_2 M_1 M_0$

2's Complement of M: $m_3 m_2 m_1 m_0$

$m = \bar{m} + 1$

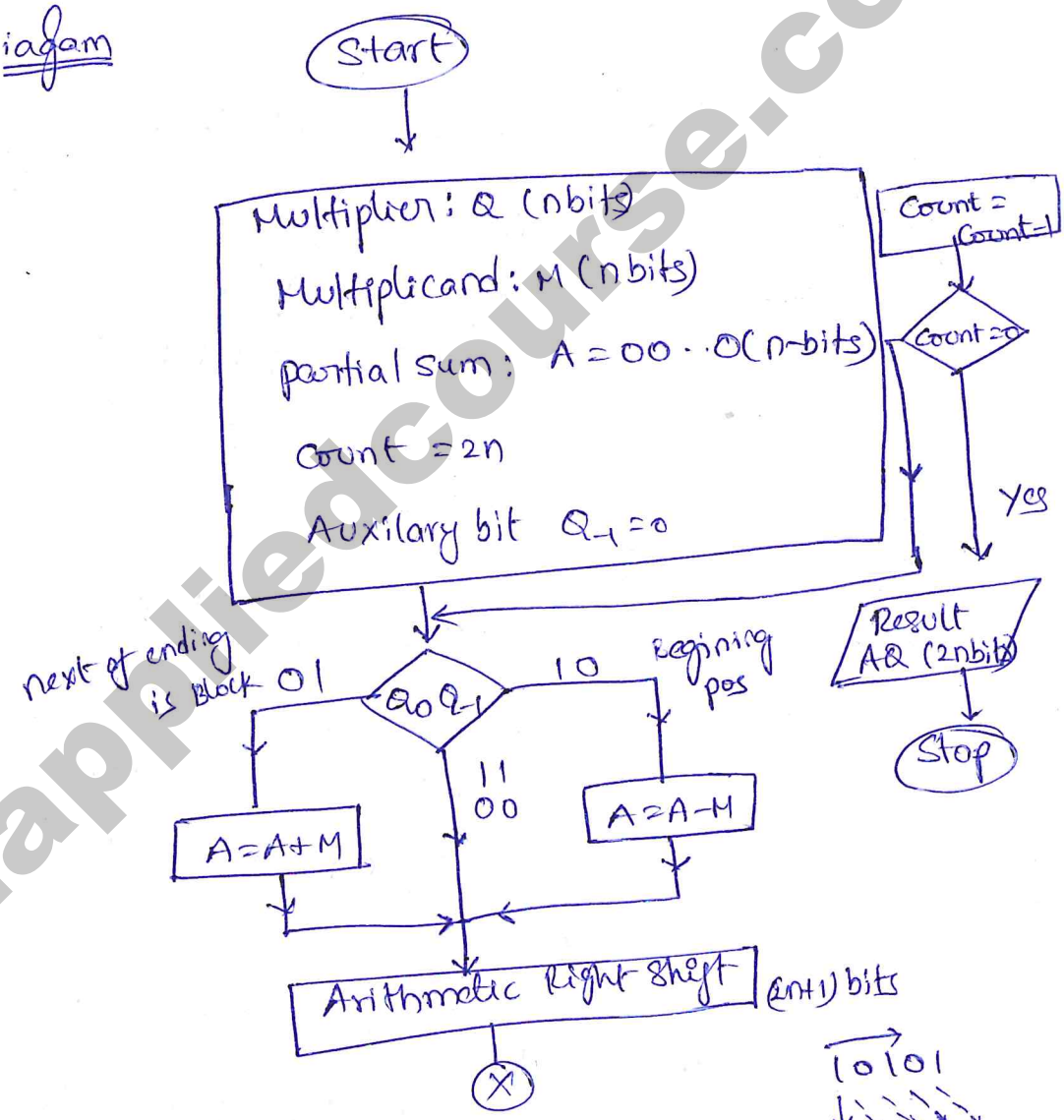
Addition; subtraction

$Q + M$ $\quad Q - M$

$\Downarrow$

$Q + \bar{M} + 1$

Flow Diagram



1 0 1 0 1

↓ ↘ ↘ ↘ ↘

1 1 0 1 0

Practice:-

M Q

$+7 * (-3) = -21$

Partial Sum A = 0000

$+7 = 0111$

$-3 = 1101$ $\quad (-8+5 = -3)$

2's Complement pattern of M = 1001 (-7)

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph:+91 844-844-0102

clock	count	partial sum	multiplier	Aux bit	Cond Q_0Q_{-1}	operation
1	4	$A_3A_2A_1A_0$	$Q_3Q_2Q_1Q_0$	Q_{-1}	10	$A = A - m$
		0000	1101	0		$= A + \bar{m} + 1$
		1001				
		1001	1101	0		Arithmatic Right Shift AQQ_{-1} (9bits)
		1100	1110	1		Count = Count-1
	3					
2	3	1100	1110	1	-01	$A = A + m$
		0111				
		0011	1110	1		Arithmatic Right Shift: AQQ_{-1} (9bits)
		0001	1111	0		Count = Count-1
	2					
3	2	0001	1111	0	10	$A = A - m$
		1001				$= A + \bar{m} + 1$
		1010	11110			
		1101	01111			Arithmatic Right Shift (9-bits)
						Count = Count-1
	1					
4	1	1101	01111	1	11	Arithmatic Right Shift AQQ_{-1}
		1110	10111			Count = Count-1
	0					

As Count = 0 Result of $(+7) \times (-3) = AQ$

$\boxed{11}10\ 1011 \Rightarrow -32+11$

$-32\ 16\ 8\ 4\ 2\ 1$

$\Rightarrow \boxed{-21}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Verification Multiplier pattern

Ph: +91 844-844-0102

$(-3)_{10}$ = 1 1 0 1

0 -1 +1 -1

Booths coding pattern

1 Addition, 2 subtractions.

Gate questions:

(1) Consider following multiplier patterns which one gives worst performance wrt Booths Algorithm.

More Arithmatic operations

S_1: 1 0 0 1 0 1 1 — 2C, 1P — 2A, 3S

S_2: 0 1 1 0 0 1 1 1 — 2C, 0P — 2A, 2S

S_3: 0 1 0 1 1 0 1 1 — 3C, 0P — 3A, 3S

S_4: 1 1 1 1 1 0 0 0 — 0C, 1P — 0A, 1S

(a) S_1 (b) S_2 (c) S_3 (d) S_4

(2) Number of arithmatic operations required for multiplier 1 0 1 1 0 1 1 1 0 1 1 1 1 with Booths Algorithm.

3C, 1P

3A, 1S

⇓

3A, 3S, 1S

(a) 4 additions, 4 Subtractions

(b) 4 Subtractions, 3 Additions

(c) 4 Additions, 3 subtractions

(d) 3 Subtractions, 4 - Additions

(3) Booths coding pattern for $-(47)_{10}$

Ex: Express -(47) in 2's Complement

$-47 = -64 + 17$

-64 32 16 8 4 2 1

1 0 1 0 0 0 1

2C + 1P

gatecse@appliedcourse.com

gateappliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

-64	32	16	8	4	2	1
1	0	1	0	0	0	1
-1	+1	-1			+1	-1

64 32 16 8 4 2 -1

$-64+32-16+2-1$

$= -48+1$

$= -47$

-1	+1	-1	0	0	+1	-1

(4) Radix-4 Booths (Algorithm) multiplication uses 3-multiplier bits for deciding Arithmatic operations. The possible operations for multiplier m, multiplicand Q are

Radix-2
$Q_0 Q_1$

(a) A+m, AM, Nop

(b) A+m, A-m

(c) A+m, A+2m, A-m, A-2M, Nop

(d) A+M, A-M, A+2M, A-2M

$Q_1 Q_0 Q_1$
Radix-4

Radix-8

$A \pm 4M, A \pm 2M, A \pm M,$ Nop

$Q_2 Q_1 Q_0 Q_{-1}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91-844-844-0102

(5) Let n-bit multiplier pattern when used with Booths Algorithm resulted P additions, Q subtractions. Identify correct relation between P, Q.

(a) $P = Q$

(b) $P = Q$, $P = Q+1$

(c) $P = Q$, $P = Q-1$ ⇐ partial is stuck

(d) $P \leq Q$

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE

III Boolean Expressions

Ph: +91 844-844-0102

3.1 Basics of Boolean Expressions:

Boolean Expressions are associated with boolean variables, boolean constants and boolean operators. Boolean expressions can be in Sum of products form (SOP) or product of sum form (POS). SOP is constructed using minterms, while POS is constructed with maxterms.

⇒ Minterms : (m) (Prime/negation/bar for 0)

Minterms are the product of the i/p variable such that total value of the product is '1'.

For example : (i) The binary pattern for minterm variables P, Q, R values as 111 and minterm will be PQR.

(ii) The binary pattern for minterm $\bar{a}b\bar{c}$ is 010

(iii) The binary pattern for minterm $a\bar{b}\bar{c}$ is 100

Thus, in short minterm is the input combination for which output is 1.

⇒ Maxterms : (M) (Prime / negation / bar for 1)

Maxterm is the sum of output variables such that total value of sum = 0 ie it is the i/p combination for which o/p is '0'

For example : If the binary variables P, Q, R with values 100 denote maxterm then maxterm expression will be: $\bar{P}+Q+R$

gateecse@appliedcourse.com

gateappliedcourse.com

APPLIED COURSE
Ph: +91-044-844-0102

Thus, for a given 3 variables A, B and C the minterm and maxterm expressions will be as follows:

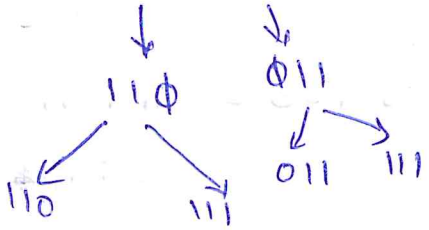
A	B	C	Minterm Expressions	Maxterm expressions
0	0	0	$A'B'C'$	$A+B+C$
0	0	1	$A'B'C$	$A+B+C'$
0	1	0	$A'BC'$	$A+B'+C$
0	1	1	$A'BC$	$A+B'+C'$
1	0	0	$AB'C'$	$A'+B+C$
1	0	1	$AB'C$	$A'+B+C'$
1	1	0	ABC'	$A'+B'+C$
1	1	1	ABC	$A'+B'+C'$

(Q) What is the binary pattern for minterms 1011.
If variables are a, b, c and d.

⇒ The minterm for 1011 ⇒ $a \cdot b' \cdot c \cdot d = ab'cd$

(Q) Minterms of 3 variable function $f(A,B,C) = ab + cb$ are?

⇒ ∵ SOP given to us $ab + bc$



∴ Minterms associated with F are } = 110, 111, 011

OR

6, 7, 3

OR

m_6, m_7, m_3.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) Identify the minterms and maxterms expressions for the following truth table:

S.No	A	B	C	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

⇒ Here, the function value, F = 1 represents the minterms while F = 0 represents the maxterms

∴ minterms : m_1, m_2, m_4, m_7

maxterms : M_0, M_3, M_5, M_6

$F(A,B,C)$ as SOP $= \Sigma(1,2,4,7)$

$= m_1 + m_2 + m_4 + m_7$

$= A'B'C + A'BC' + AB'C' + ABC$

$F(A,B,C)$ as POS $= \Pi(0,3,5,6)$

$= M_0 \cdot M_3 \cdot M_5 \cdot M_6$

$= (A+B+C) \cdot (A+B'+C') \cdot (A'+B+C') \cdot (A'+B'+C)$

NOTE: Boolean expression doesn't change whether it is written SOP or POS.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.2 Canonical form and Standard form:

Canonical form: Each term of boolean expression contain all input variables. either in true form or in complement form.

Example: (i) $F(A,B,C) = A'B'C' + ABC$ ⇒ it is Canonical SOP expression (CSOP)

(ii) $F(A,B,C,D) = A'BCD + AB'C'D + ABCD$ ⇒ It is also a Canonical SOP expression (CSOP)

Standard form: If there exists at least one term that does not contain all variables.

Example: (i) $F(A,B,C) = A' + ABC$ ⇒ It is standard SOP expression (SSOP)
(B and C are missing)

(ii) $F(A,B,C,D) = AB + BC + A'B'C'D'$ ⇒ It is standard SOP expression (SSOP)

(Q) Which of the following functions are in Canonical form?

$F_1(A,B,C) = (A+B+C)$

$F_2(A,B,C) = AB + BC + AC$

$F_3(A,B,C) = A'BC + AB'C + ABC'$

$F_4(A,B) = A'B' + AB$

(a) F_1, F_2, F_3 (b) F_2, F_3, F_4 (c) F_1, F_3, F_4 (d) ALL

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE Ph: +91 844-844-0102

Here, F_1 is in Canonical Pos form

F_2 is in standard SOP form

F_3 and F_4 are in canonical SOP form

thus correct option is (C) F_1, F_3 and F_4

NOTE- For function $F_1(A,B) = A+B$

↳ it is standard SOP But it is canonical Pos

$F_2(A,B) = AB$

↳ it is standard Pos but it is canonical SOP

3.3 Example problems on minterms and maxterms:

(Q) How many self dual functions are possible with 2, 3, 4 boolean variables

(a) 2, 4, 8

(b) 4, 8, 16

(c) 4, 16, 256

(d) 8, 16, 512

⇒ Since # self dual functions for n boolean variables is given as $= 2^{2^{n-1}}$

$n = 2 \Rightarrow 2^{2^1} = 2^2 = 4$

$n = 3 \Rightarrow 2^{2^2} = 2^4 = 16$

$n = 4 \Rightarrow 2^{2^3} = 2^8 = 256$

∴ (C) is correct option.

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(9) With respect to the following statements of 3 variables identify the correct ones

S_1 : $F(A, B, C) = \Sigma(0, 1, 2, 3, 4)$

S_2 : $F(A, B, C) = \Sigma(0, 1, 2, 3)$

S_3 : $F(A, B, C) = \Sigma(1, 2, 4, 7)$

	S_1	S_2	S_3
(a)	Self-dual	Orthogonal	Self-dual
(b)	non-orthogonal and non-self dual	Self-dual	Orthogonal
(c)	non-orthogonal and non-self dual	Self-dual	Self dual
(d)	Orthogonal	Orthogonal	Self dual

$\Rightarrow$

Self dual function $\Rightarrow$ (i) no. of minterms = no. of maxterms

(ii) No mutually exclusive terms

ie $F(A, B, C) = F(A', B', C')$

In S_1: (i) minterms = 5 $\therefore$ maxterms = 3

$\therefore$ it is not self dual

In S_2: (ii) # minterms = # maxterms = 4

(iii) There should be no mutually exclusive terms ie following pairing should not be there

(0,7) (1,6) (2,5) (3,4)

$\therefore S_2 = \Sigma(0, 1, 2, 3)$ $\therefore$ no mutually exclusive term is present

$\therefore$ it is self-dual

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

~~Thus~~

Thus, we can eliminate option (d)

In S_3: (i) # minterms = # maxterms = 4

(ii) No mutual exclusive term

∴ it is also self dual

Hence, correct option is (c)

Gate 2015 set 3

(9) Given $F = P'+QR$ where F is a function of 3 boolean variables P, Q and R and $P' = !P$.

Consider the following statements

S_1: $F = \Sigma(4,5,6)$

S_2: $F = \Sigma(0,1,2,3,7)$

S_3: $F = \Pi(4,5,6)$

S_4: $F = \Pi(0,1,2,3,7)$

Which of the following are true based on the 4 statements.

	S_1	S_2	S_3	S_4
(i)	F	T	T	F
(ii)	T	F	F	T
(iii)	F	F	T	T
(iv)	T	T	F	F

⇒ Here S_1 can never be equal to S_3
S_2 " " " " " S_4
S_3 " " " " " S_4
S_1 " " " " " S_2
} all have common terms

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Option (ii) is eliminated as $S_3 \neq S_4$. Similarly (iv) is eliminated as $S_1 \neq S_2$.

$S_1 \equiv S_4$ and $S_2 = S_3$ $\Rightarrow$ based on this both options (a) and (b) could be correct

Now, in order to eliminate we will consider the expression

$$F(P, Q, R) = P' + QR$$

minterms $P' \Rightarrow P' \Rightarrow 0\,\phi\,\phi \Rightarrow$ 000, 001, 010, 011 (P Q R; 0, 1, 2, 3)

minterms $QR \Rightarrow QR \Rightarrow \phi\,1\,1 \Rightarrow$ 011, 111 (P Q R; 3, 7)

So S_1 is False as it dont have 0, 1, 2, 3, 7

Hence S_2 and S_3 are true and S_1 and S_4 are False.

∴ Correct option is (i)

(9) $F_1(A, B, C) = \leq (0, 2, 4, 7)$

$F_2(A, B, C) = \leq (1, 4, 7)$

$F(A, B, C) = \leq (4, 5, 7)$

Let $F = F_1 F_2 + F_3$; how many functions are possible for $F_3(A, B, C)$

(a) 1

(b) 2

(c) 4

(d) 8

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$\Rightarrow$ $F_1 \cdot F_2 \Rightarrow$ will give the common minterms in both F_1 and F_2

$\Rightarrow \Sigma(4,7)$

$\therefore F_1 \cdot F_2 + F_3 = F$

$\downarrow$

$\Sigma(4,7) + ? = \Sigma(4,5,7)$

Possible solution of F_3 are $\Rightarrow$ $\Sigma(5,)$, or
$\Sigma(4,5)$, or
$\Sigma(4,5,7)$, or
$\Sigma(5,7)$

Hence, 4 possible functions are there for F_3.

(Q) $F_1(A,B,C) = \Sigma(0,4,5)$, $F_2(A,B,C) = \Sigma(0,6)$

$F_1 \oplus F_2 \oplus F_1 \cdot F_2 = ?$

(a) $\Sigma(0,4,5)$
(b) $\Sigma(0,6)$
(c) $\Sigma(0,4,5,6)$
(d) $\Sigma(0)$

$\Rightarrow$ $A \oplus B \oplus AB = A + B$

$\therefore$ The result will be including all the minterms of F_1 and F_2

$F_1 + F_2 = \Sigma(0,4,5,6)$

Hence, correct option is (c) //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.4 Characteristics of Boolean Expression:

- (i) Duality (used for indirect proof of exp AND→OR, 0→1 and vice versa but inversion remains same)
- (ii) Complementation (AND→OR, 0→1 and vice versa variable are replaced by their complements)

Duality: For every boolean expression there exists dual expression

Example 1: Let $F(A, B) = A'B + AB'$

$F_{dual} = (A' + B)(A + B')$ (AND→OR, OR↔AND but inversion of variable is not done)

$F' = (A'' + \overline{B})(A' + (B')')$

$= (A + B')(A' + B)$

Example 2:

$F(P, Q, R) = P' + QR$

$F_{dual} \Rightarrow F_D(P, Q, R) = P'.(Q + R)$

$F_{complement} \Rightarrow F'(P, Q, R) = P(\overline{Q} + \overline{R})$

Example 3:

$F(A, B) = A + 0$

$F_D(A, B) = A \cdot 1 = A$

$F'(A, B) = A' \cdot 1 = A'$

Note: No. of variables will remain same in the expression.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.5 Boolean Expression characteristics : Duality :

Let a boolean function F and its dual is F_D then if F is having boolean property P, its dual indeed will have same property.

F is independent of variable x. The FDual is also independent of x.

Why we need dual functions ?

* Some times dual function realization require less hardware than original
* Duality is used for converting boolean realization across +ve logic and -ve logic systems
* Indirect proof for some situations.

How to get dual functions ?

* AND operation is replaced with OR viceversa
* Boolean constant '1' is replaced with '0' vice versa.
* Inversion remains unchanged.

In terms of expression, it could be given as -

$$F(x_1, x_2, x_3 \cdots x_n, 0, 1, +, \cdot)$$

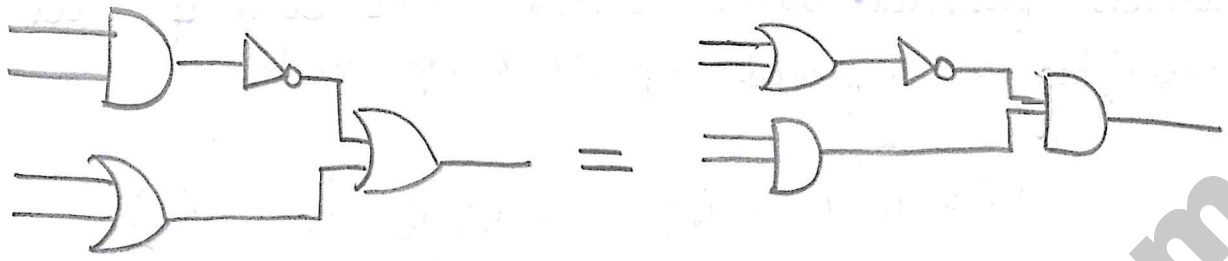
$$F_D = F(x_1, x_2, x_3 \cdots x_n, 1, 0, \cdot, +)$$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

In terms of logic gates it could be given as -

Ex 1 -



$F \Rightarrow F_{Dual}$

Ex 2 - $F_1 = x'y + w'z \Rightarrow F_D = (x'+y)\cdot(w'+z)$

NOTE -

Based on the relation between function F and its dual function F_D, we can classify all the boolean functions into one of the following groups:

(i) Self dual $\Rightarrow F = F_D$ (Function = dual)
(ii) Orthogonal $\Rightarrow F = F_D'$ or $F_D = F'$
(iii) Non-Self dual and Non-orthogonal
$F \neq F_D$ and $F \neq F_D'$

3.6 Boolean Expressions Characteristic: Self Duality:

Properties of self dual function:

(i) # minterms must be equal to the # maxterms
(ii) And, there should not be any mutually exclusive terms.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

The mutually exclusive terms is the one in which function value remain same even if all variables are replaced by its complements.

i.e.

$$F(x_1, x_2, x_3 \cdots x_n) = F(x_1', x_2', x_3' \cdots x_n')$$

NOTE-

Self dual functions with n-boolean variables is equal to $\boxed{2^{2^{n-1}}}$

∵ with n variables we can have 2^n minterms out of which half are minterms and half are maxterm ie 2^{n-1} and by replacing this with 2^{n-1} slots with 0 and 1 individually, we can have $2^{2^{n-1}}$ functions which are self dual.

(Q) No. of self dual function with 3 boolean variables?

⇒ $2^{2^{n-1}}$ function for n variables.

∴ $n = 3$

$$\therefore 2^{2^{3-1}} = 2^{2^2} = 2^4 = 16 \text{ //}$$

(Q) Which of the following are self dual?

S_1: $F_1(A, B, C) = \Sigma(0, 1, 2, 3)$

S_2: $F_2(P, Q, R) = \Pi(2, 3, 6, 7)$

S_3: $F_3(X, Y, Z) = \Sigma(0, 1, 4)$

S_4: $F_4(L, M, N) = \Sigma(2, 3, 4, 7)$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(a) S_1, S_2, S_3

(b) S_2, S_3, S_4

(c) S_1, S_2

(d) S_3, S_4

⇒ ∵ In S_3 ⇒ # minterms ≠ # maxterms

∴ we can eliminate all the options with S_3

∵ only option (c) is left ∴ it is the correct answer.

Moreover this, S_4 ~~Can~~ is having (3, 4) together though they are mutually exclusive.

(9) Identify the valid statement/s :

S_1 : Sum, Carry function of full adder are self dual

S_2 : Difference, borrow function of full adder are self dual

S_3 : For even no. of i/p's EX-OR become orthogonal

S_4 : Self duality is used for analyzing super Conductivity

(a) S_1, S_2, S_3

(b) S_2, S_3, S_4

(c) S_1, S_3, S_4

(d) All

⇒ All 4 statements are true, thus correct option is (d)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.7 Boolean expression Characteristics: Complementation and Demorgan laws:

Complementation: F is the boolean function, its complement F' is derived by:

* Replacing variables by its complement
* AND with OR vice versa
* Boolean constant '1' with '0' and vice versa.

Example 1 -

$$F(A,B) = A'B' + AB$$
$$F'(A,B) = (A+B)(A'+B')$$

Example 2 -

$$F(A,B) = A+0 = A$$
$$F'(A,B) = A' \cdot 1 = A'$$

De-Morgan laws:

(i) If $F(A,B) = A+B$, then
$F'(A,B) = A'B'$
(NOR: Bubbled AND)

ie if $\overline{A+B} = A'B'$ OR $(A+B)' = A'B'$

(ii) if $F(A,B) = AB$, then
$F'(A,B) = A'+B'$
(NAND: bubbled OR)

ie. if $\overline{AB} = A'+B'$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Note:

✳ Dual in terms of complementation

$$F_D = F'(x_1', x_2' \cdots x_n')$$

If all variables of complemented functions are complemented, then it becomes dual function.

Proof -

LHS

$F(A,B) = A + B$

$F'(A,B) = A' \cdot B' = A'B'$

$F'(A', B') = (A')' \cdot (B')' = AB$

RHS

$F(A,B) = A + B$

$F_D(A,B) = A \cdot B$

$\therefore$ LHS = RHS

as $F'(A',B') = F_D(A,B)$ and vice versa

(Q) Let $F(x_1, x_2, x_3 \cdots x_n)$ and its complement $F'(x_1, x_2 \cdots x_n)$ then how many boolean functions are possible such that

$$F(x_1, x_2, \cdots x_n) = F'(x_1', x_2' \cdots x_n')$$

⇒ As we know if variable of complemented functions are complemented then it becomes dual

ie $F(x_1, x_2 \cdots x_n) = F'(x_1', x_2' \cdots x_n') = F_D$

$\therefore$ # self-dual functions $= 2^{2^{n-1}}$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

NOTE:

* Complementation in terms of Duality

$$F' = F_D(x_1', x_2', \ldots x_n')$$

If all variables of dual function are complemented then it becomes complemented function.

Proof-

LHS

$F(A, B) = A + B$

$F_D(A, B) = A \cdot B = AB$

$\downarrow$

$F_D(A', B') = A'B'$

RHS

$F(A, B) = A + B$

$F'(A, B) = A' \cdot B'$

$\therefore$ LHS = RHS

ie $F' = F_D(A', B')$

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE

Ph: +91 844-844-0102

3.8 Representation of Boolean Expressions:

With n-boolean variables, we can have 2^{2^n} boolean functions.

Ex- $n = 2$

then, no. of boolean functions $= 2^{2^2} = 2^4 = 16$

These 16 functions will be as follows:

A	B	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
		NULL function	AB	AB'	A	$A'B$	B	$A \oplus B$	$A+B$	$A'B'$	$A \odot B$	B'	$A+B'$	A'	$A'+B$	$A'+B'$	Tautology

Representation: $^2C_0 + {}^2C_1 + {}^2C_2 + {}^2C_3 + {}^2C_4 = 2^{2^2} = 2^4 = 16$

$${}^4C_0 + {}^4C_1 + {}^4C_2 + {}^4C_3 + {}^4C_4 = 2 \cdot {}^4C_0 + 2 \cdot {}^4C_1 + {}^4C_2$$
$$= 2 \cdot 1 + 2 \cdot 4 + 6$$
$$= 16$$

(S) How many boolean functions are possible with 3-boolean variables such that they contain either 3 minterms or 4 minterms?

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

⇒ Since, with 3 variables we can have total $2^{2^3} = 2^8 = 256$ functions

but with 3 minterms ~~and~~ at 4 minterm will be

$${}^8C_3 + {}^8C_4 = 56 + 70 = 126$$

(Q) For at most 3 minterms for 3 boolean variables what will be the total possible functions?

⇒ with at most 3 minterms

$${}^8C_0 + {}^8C_1 + {}^8C_2 + {}^8C_3 = 1 + 8 + \frac{8\times7}{2} + \frac{8\times7\times6}{3\times2}$$

$$= 1 + 8 + 28 + 56$$

$$= \underline{93} \text{ functions.}$$

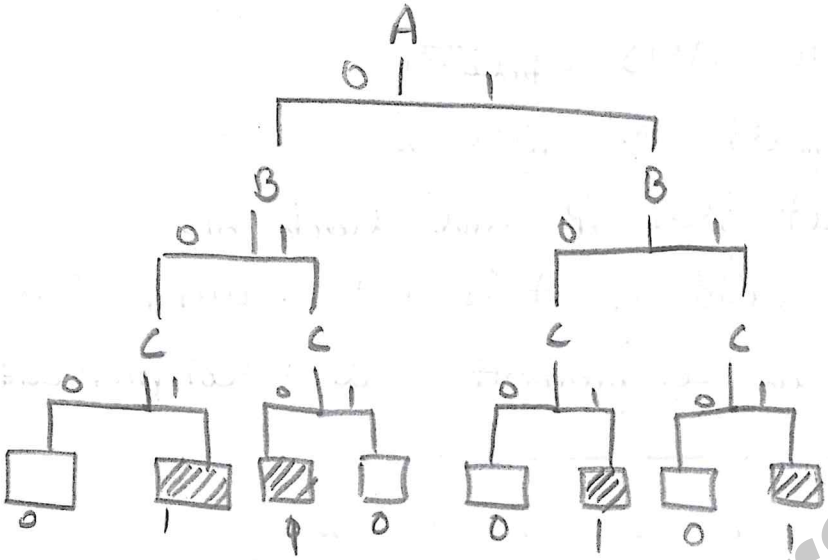
(Q) # possible functions for 3 boolean variables ~~and~~ having exactly 5 minterms?

⇒ $${}^8C_5 = {}^8C_3 = \frac{8\times7\times6}{3+2} = 56$$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

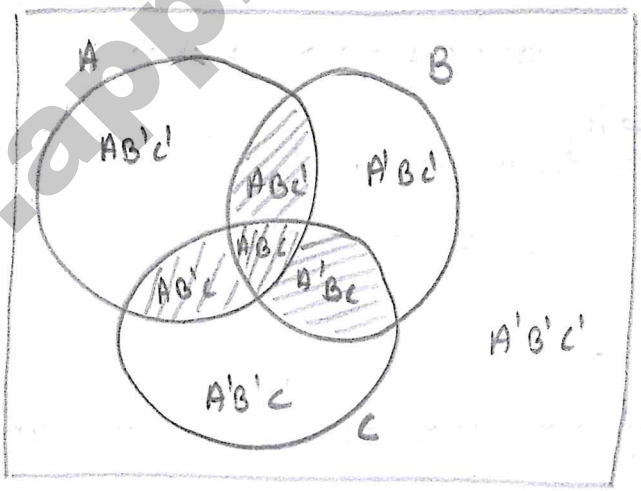
Binary tree representation of boolean expression -



here minterms are evaluated/backtracked : 001, 010, 101, 111

$= \Sigma(1, 2, 5, 7)$

Venn Diagram of boolean expression -



minterms are represented by shading Venn region.

Here minterms are the one which are covered/shaded

ie $F(A, B, C) = ABC' + AB'C + ABC + A'BC$

$= 110 + 101 + 111 + 011$

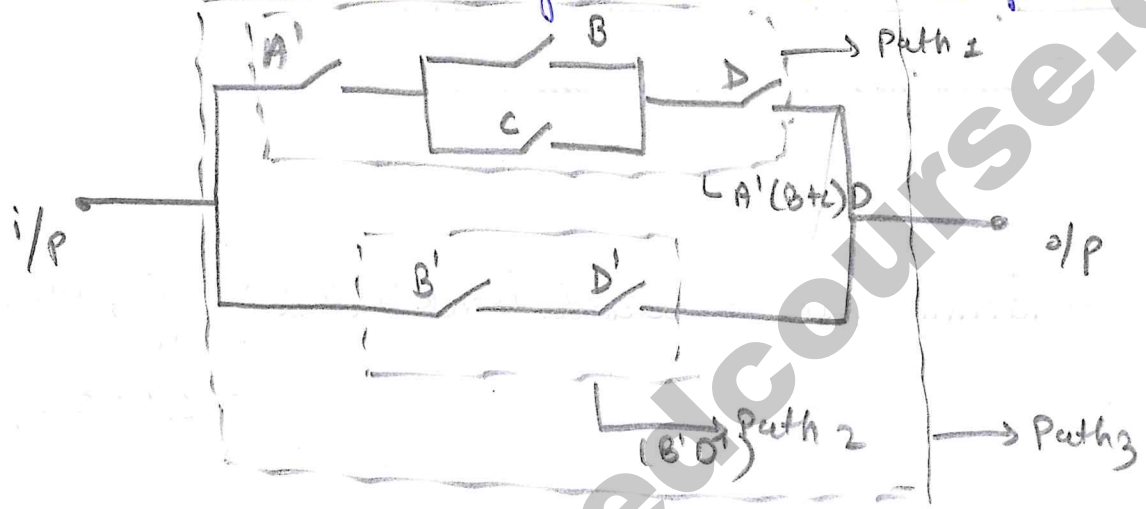
$= \Sigma(3, 5, 6, 7) = \Pi(0, 1, 2, 4)$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Contact Network representation of Boolean Expression:

- Serial path AND operation
- Parallel path OR operation
- Path start from i/p and reach o/p
- Path is valid if it is not having same literal in complement and uncomplement form.



The o/p will be like

$$A'(B+C)D + B'D'$$

∴ o/p we get on expanding it are:

$$= A'BD + A'CD + B'D'$$

3.9 Example problems on Boolean representations

(Q) How many maxterms are possible with n boolean variables

(i) 2^n
(ii) 2^{n-1}
(iii) 2^{2^n}
(iv) $2^{2^{n-1}}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

⇒ With n boolean variables, we can have 2^n rows and if all the rows are having 0 for any function ~~F(n)~~ F (maxterms) than max possible maxterms will be 2^n

∴ Correct option is 2^n

(Q) How many boolean functions are possible with 4 boolean variables?

(i) 16
(ii) 65536
(iii) 4096
(iv) 32768

⇒ ∵ with n boolean variables, we can have total 2^{2^n} functions

∴ for n = 4 ⇒ $2^{2^4} = 2^{16} = 2^6 \times 2^{10} = 1024 \times 64$
$= 65536$ //

(Q) How many boolean functions with 3-boolean variables such that the no. of minterms are atmost 5

(i) ${}^8C_0 + {}^8C_1 + {}^8C_2 + {}^8C_4$
(ii) ${}^8C_6 + {}^8C_7 + {}^8C_8$
(iii) ${}^8C_0 + {}^8C_1 + {}^8C_3 + {}^8C_4 + {}^8C_5$
(iv) ${}^8C_{2^8} - ({}^8C_6 + {}^8C_7 + {}^8C_8)$

⇒ Since atmost 5 is asked ∴ answer will be

${}^8C_0 + {}^8C_1 + {}^8C_2 + {}^8C_3 + {}^8C_4 + {}^8C_5$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Since ${}^8C_2$ is no included in (iii) optim

∴ we will check for optim (iv)

$$\therefore 2^8 = {}^8C_0 + {}^8C_1 + {}^8C_2 + {}^8C_3 + {}^8C_4 + {}^8C_5 + {}^8C_6 + {}^8C_7 + {}^8C_8$$

$$2^8 - ({}^8C_8 + {}^8C_7 + {}^8C_8) = {}^8C_0 + {}^8C_1 + {}^8C_2 + {}^8C_3 + {}^8C_4 + {}^8C_5$$

Hence, (iv) is correct //

3.10 Example problems on Generalized boolean functions and representation:

(8) How many boolean function are possible for 3 - boolean variables such that

$$f(x,y,z) = F(x,y,z')$$

(a) 8
(b) 16
(c) 128
(d) 256

⇒

$f(x,y,z) = F(x,y,z')$

x	y	z	
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

∴ 8 boolean function are reduced to 4

$\therefore 2^4 = 16 = 2^{2^n}$ //

for $F(x,y,z) = F(x,y,z')$

∴ (b) is correct //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(9) How many binary functions are possible with 2-trinary Variables.

(a) 2^{2^3}
(b) 2^{2^2}
(c) 2^{3^2}
(d) 3^{3^2}

$\Rightarrow$ The no. of n-ary function with q variables of R-valued algebra are n^{R^q}

$q \rightarrow$ no. of variable
$R \rightarrow$ nature of variable
$n \rightarrow$ nature of function

here we have 2 trinary variable

$\therefore q = 2$
$R = 3$
$n = 2$

$= 2^{3^2}$ //

3.11 Boolean laws Overview, Identity law:

Boolean laws:

* **Identity law:** Identity operation on identical variable give unique result

$A + A + A = A$

$A \cdot A \cdot A = A$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* Commutative: Order of variables does not change functionality

$A + B = B + A$

dual: $AB = BA$

It is useful while realizing IC's (Integrated chips) as changing the order of variable does not change the o/p.

* Associative: Order of variables does not change functionality.

$A + (B + C) = (A + B) + C$

dual: $A \cdot (B \cdot C) = (A \cdot B) \cdot C$

* Distributive law: It is excessively used in minimization

$A + BC = (A + B)(A + C)$

dual: $A \cdot (B + C) = A \cdot B + A \cdot C$

All these rules/laws are used for the minimization of boolean expressions.

(8) Given that $A \$ B = A + B'$. Here $ is a operation performed between A and B resulting in $A + B'$. $ is $ follows identity law or not?

⇒ In order to check identity law, lets replace the A and B with P.

$(((P \$ P) \$ P) \$ P) \$ P) = ?$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$\therefore A \$ B = A + B'$

$\therefore P \$ P = P + P' = 1$

$\therefore (((((P \$ P) \$ P) \$ P) \$ P) \$ P)$

$P + P' = 1$

$(((1 \$ P) \$ P) \$ P) \Rightarrow ((1 \$ P) \$ P) \$ = 1 \$ P$

$= 1 + P'$

$= 1$

$\therefore$ we got single answer 1 $\therefore$ it follows identity law.

(9) Check whether EX-OR follows identity law or not ?

$\Rightarrow (((((x \oplus x) \oplus x) \oplus x) \oplus x) \oplus x) = (((0 \oplus x) \oplus x) \oplus x)$

$= (x \oplus x) \oplus x$

$\therefore$ it is not giving us single answer (we are getting two answers 0 and x)

$\therefore$ Ex-OR is not following identity law.

$\therefore$ Ex-NOR is dual of Ex-OR $\therefore$ it also do not follows identity law.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Similarly we can check for NAND and NOR also and they also do not follows identity law.

3.12 Boolean laws - Commutative, Associative, Distributive laws:

* Commutative law:

$$A + B = B + A$$

dual: $A\,B = B\,A$

If the boolean function remain same even if variables are permuted

ie $F(A, B, C) = F(B, C, A) = F(C, A, B) = F(A, C, B) = F(B, A, C)$

Example:

$$F = A'BC + AB'C + ABC' + ABC = \Sigma(3, 5, 6, 7)$$

$$F = ABC + A'B'C = \Sigma(0, 7)$$

In general, AND, OR, NAND, NOR, XOR, X-NOR, follows the commutative law.

* Associative law:

$$A + (B + C) = (A + B) + C$$

dual: $A \cdot (B \cdot C) = (A \cdot B) \cdot C$

NOTE: AND, OR, EX-OR, EX-NOR follows associative law while NAND, NOR doesnot follow associative law.

Proof for Ex-OR -

$$A \oplus (B \oplus C) = (A \oplus B) \oplus C$$

LHS $\Rightarrow$ $A \oplus B \oplus C = A \oplus (B'C + BC')$

$= A'(B'C + BC') + A(B'C' + BC)$

$= A'B'C + A'BC' + AB'C' + ABC$

$= (A \oplus B)'C + C'(A \oplus B)$

$= (A \oplus A')' \oplus C$ By (1)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* Distributive law:

$$A+(B\cdot C) = (A+B)(A+C)$$

dual: $A\cdot(B+C) = AB\ AC$

There are 2 forms of distributive law

(i) Combining law — $AB + AB' = A$

dual: $(A+B)(A+B') = A$

(ii) Absorption law — $A + AB = A$

dual: $A(A+B) = A$

ex- $(P+Q)(P+R)(P+S)(P+T) = ?$

⇒ Acc. to distributive law (∵ P is common)

$P+(QRST)$

ex- $(P+Q)(P+R)(P+S)(P+R') = ?$

⇒ Acc to combining law of distributive law

$P+QRSR' = P+0 = P$ //

ex- $x+x'y = x+y$ How?

⇒ ∵ On applying distributive law

$(x+x')(x+y) = (x+y)\cdot 1 = (x+y)$ //

dual:- $x(x'+y) = xy$ //

ex- $x+abc = ?$

⇒ $(x+a)(x+b)(x+c)$ //

ex- $x'+yzx = x'+yz$ How?

⇒ $(x'+y)(x'+z)(x'+x) = (x'+y)(x'+z)$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) Which of the following is commutative, Associative but not distributive?

(a) AND
(b) NAND
(c) Ex-OR
(d) OR

⇒ ∵ AND, OR follows all the laws ∴ we can remove them from options.

NAND is commutative but not associative.

Hence, correct option is Ex-OR (Ex-OR is commutative and associative but not distributive)

Proof -

$A \oplus (B \oplus C) = (A \oplus B) \oplus (A \oplus C)$

here, Ex-OR is distributed on Ex-OR

let $A = 1$, $1 \oplus (B \oplus C) = (B \oplus C)'$ (LHS)

$(1 \oplus B) \oplus (1 \oplus C) = B' \oplus C'$ (RHS)

∴ LHS ≠ RHS

∴ Ex-OR does not follow distributive law.

(Q) $(A'+B'+C')(A'+B'+C)(A'+B+C')(A'+B+C) = ?$

(a) A'
(b) A
(c) $(A'+B')$
(d) $(A'+B'+C')$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$\Rightarrow (A'+B'+C')(A'+B'+C)(A'+B+C')(A'+B'+C')$

$(A'+B')$ $\qquad$ $C'+(A'+B)(A+B')$

$\Rightarrow (A'+B')(A'+B+C')(A'+B'+C')$

$= (A'+B'C')(A'+B'+C')$

$= (A'+B')(A'+C')(A'+B'+C')$

$= (A'+B')(C'+A'(A+B'))$

$= (A'+B')(C'+A'B')$

$= A'+A'C'+A'B'+B'C'(A'+B+C)$

$= A'+A'B+A'C+A'C'+A'BC'+A'C\cdot C+A'B'+A'B'B+A'B'C+A'B'C'+B'C'B+B'C'C$

$= A'(1+B+C+C'+BC'++B'+B'C+B'C')$

$= A'$

(Q) $(P'+Q+R')(P'+Q'+R)(Q'+R')=?$

$\Rightarrow (P'+(Q+R')(Q'+R))(Q'+R')$

$= P'(Q'+R')+(Q'+R')(Q+R')(Q'+R)$

$= P'Q'++P'R'+R'Q'$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) $a + a'b + a'b'c + a'b'c'd + a'b'c'd'e = ?$

$\Rightarrow a + \underbrace{a'b + a'b'c + a'b'c'd + a'b'c'd'e}_{\text{on taking } a' \text{ common}}$ $\left(\because x + x'y = x + y\right)$

$\Rightarrow a + b + \underbrace{b'c + b'c'd + b'c'd'e}_{\text{on taking } b' \text{ common}}$

$\Rightarrow a + b + c + \underbrace{c'd + c'd'e}_{\text{on taking } c' \text{ common}}$

$\Rightarrow a + b + c + d + d'e$

$\Rightarrow a + b + c + d + e$

(Q) $w(w' + x' + y')(w' + xy) = ?$

$\Rightarrow \because w(w' + A)(w' + A') = w \cdot w'$ (as A and A' will get cancel)

$= w(w' + x' + y')(w' + xy)$

$= w \cdot w'$

$= 0$

(Q) Valid statements

$A \oplus B = Z$

$S_1 : Z \oplus B = A$

$S_2 : Z \oplus A = B$

$S_3 : Z' \oplus A \oplus B = 1$

(a) Only S_1 (b) Only S_2 (c) S_1 and S_2 (d) S_1, S_2 and S_3

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

⇒ For S_1 :

$\because A \oplus B \oplus B = A \oplus 0 = A$

$\therefore Z \oplus B = A$ is valid

For S_2 :

$\because Z \oplus A = A \oplus B \oplus A = A \oplus A \oplus B = 0 \oplus B = B$

$\therefore Z \oplus A = B$ is valid

For S_3 :

$\because Z' \oplus A \oplus B = Z' \oplus \underbrace{A \oplus B}_{Z} = Z' \oplus Z = 1$

$\therefore$ it is also valid

$\therefore$ (d) is correct option //

(Q) Given $P = A \oplus B$. what is the output of the following?

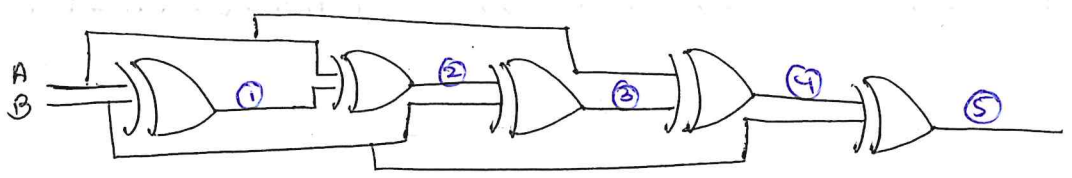
$P \oplus A \oplus P \oplus B \oplus A \oplus P \oplus A \oplus P = ?$

⇒ On expanding the expression we will get

$\Rightarrow A \oplus B \oplus A \oplus A \oplus B \oplus B \oplus A \oplus A \oplus B$

$\Rightarrow A \oplus B$ ($\because A \oplus A = 0$, $B \oplus B = 0$)

(Q)



⇒ At (1) ⇒ $A \oplus B$

At (2) ⇒ $A \oplus B \oplus A = 0 \oplus B = B$

At (3) ⇒ $B \oplus B = 0$

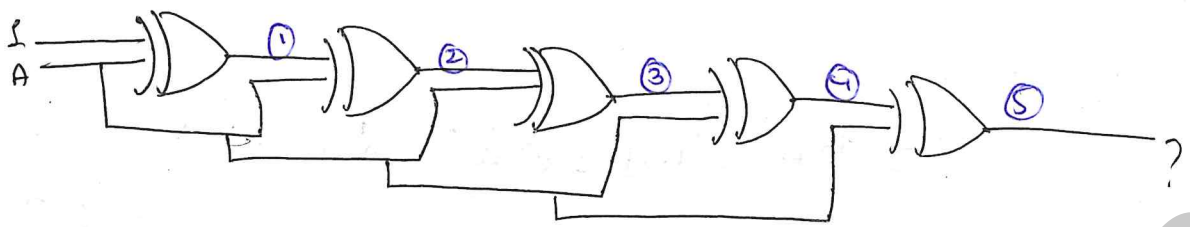
At (4) ⇒ $0 \oplus A = A$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

At (5) ⇒ $A \oplus B = A \oplus B$.

(Q)



⇒ At (1) ⇒ $1 \oplus A = A'$

At (2) ⇒ $A' \oplus A = 0$

At (3) ⇒ $0 \oplus A = A$

At (4) ⇒ $A \oplus A = 0$

At (5) ⇒ $0 \oplus A = A$

3.13 Compensation law, Shanon's Expansion Theorem:

* **Compensation law:**

It is used to identify and remove redundant term from the expression.

In SOP, if a variable is associated with a term P_1 and its complement is associated with term P_2 then P_1P_2 term in SOP is redundant

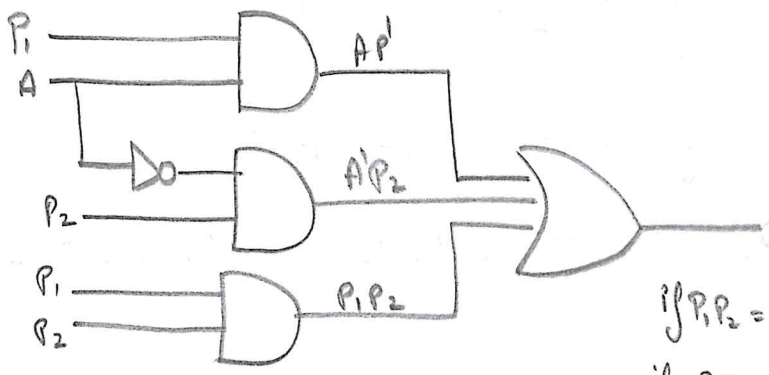
i.e $AP_1 + A'P_2 + P_1P_2 = AP_1 + A'P_2$

dual: $(A+P_1)(A'+P_2)(P_1+P_2) = (A+P_1)(A'+P_2)$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Circuit diagram of SOP form:



if $P_1P_2 = 0 \Rightarrow AP_1 + A'P_2$

if $P_1P_2 = 1 \Rightarrow A + A' = 1$
$(P_1 = P_2 = 1)$

Hence, presence or absence of term P_1P_2 didnot make any difference in the result.

ex- $\omega x + \omega yz + xyz = \omega' x + \omega yz$.

* Shanon's Expansion Theorem:

It is useful for expressing higher variable boolean function as the combination of lower variable functions

Ex- $F(x_1, x_2, x_3 \cdots x_{k-1}, x_k, x_{k+1} \cdots x_n)$ } n variable function

Expansion with respect to variable 'k'

$f(x_1, x_2 \cdots x_{k-1}, 0, x_{k+1}, x_{k+2} \cdots x_n) \cdot x_k' +$
$f(x_1, x_2 \cdots x_{k-1}, 1, x_{k+1}, x_{k+2} \cdots x_n) \cdot x_k$

} (n+1) variable function
It is an SOP of 2, (n+1) variable function.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE Ph: +91 844-844-0102

Ex- Expansion of above function wrt x_1

$\Rightarrow f(0, x_2, x_3 \cdots x_n) * x_1' + f(1, x_1, x_2 \cdots x_n) * x_1$

Similarly, we can expand it wrt x_n

$\Rightarrow f(x_1, x_2 \cdots 0) * x_n' + f(x_1, x_2 \cdots x_{n-1}, 1) * x_n$

(Q) Let a 2-variable function $f(x, y)$ is expressed as

$a_0 x'y' + a_1 x'y + a_2 xy' + a_3 xy$

where $a_0 = f(0,0)$

$a_1 = f(0,1)$

$a_2 = f(1,0)$

$a_3 = f(1,1)$

If $f(P, Q) = P' + Q$ then Boolean constant value is?

	a_0	a_1	a_2	a_3
(a)	0	0	0	0
(b)	0	1	0	0
(c)	1	0	1	1
(d)	1	1	0	1

$\Rightarrow$ Given function $f(P, Q) = P' + Q$

$\therefore a_0 = f(0,0) = (0)' + 0 = 1 + 0 = 1$

$a_1 = f(0,1) = (0)' + 1 = 1 + 1 = 1$

$a_2 = f(1,0) = (1)' + 0 = 0 + 0 = 0$

$a_3 = f(1,1) = (1)' + 1 = 0 + 1 = 1$

$\therefore a_0 = 1, a_1 = 1, a_2 = 0, a_3 = 1$

Hence, option (d) is correct //

gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.14 Solved problems:

(Q) Which of the following POS is in standard form

$F_1(A, B, C) = (A' + B' + C)(A + B + C)$

$F_2(A, B, C) = (A + B)(A' + B')$

$F_3(A, B, C) = A'B'(A + B + C)$

$F_4(A, B, C) = A(B' + C')$

(a) F_1, F_2, F_3

(b) F_2, F_3, F_4

(c) F_2, F_4

(d) Only F_4

$\Rightarrow$ here, F_1 is having all variables in all of its terms

∴ it is canonical POS

F_2 is not having C in its terms

∴ it is standard POS

F_3 is neither in SOP nor POS

F_4 is not having all variables in all of its terms

∴ it is in standard POS

∴ (c) is correct option //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) The minterms of three variable function $F(P, Q, R) = P + Q'R'$

(a) m_0, m_4, m_5, m_6, m_7

(b) m_0, m_4, m_5, m_6

(c) m_0, m_1, m_2, m_3, m_4

(d) m_0, m_1, m_2, m_3

$\Rightarrow$

For term P $\Rightarrow$ P Q R
1 ϕ ϕ

The minterms will be: 100 → 4, 101 → 5, 110 → 6, 111 → 7

For term $Q'R'$ $\Rightarrow$ P Q' R'
ϕ 0 0

The minterms will be: 100 → 4, 000 → 0

∴ minterms will be m_0, m_4, m_5, m_6, m_7

Hence, (a) is the correct option //

The expression will be:

$\Rightarrow P'Q'R' + PQ'R' + PQ'R + PQR' + PQR$ //

The maxterms are:— m_1, m_2, m_3

expression:— $(P' + Q' + R')(P + Q' + R)(P + Q' + R')$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(8) Identify the equivalency of 2- Three variable function $F_1(a,b,c)$, $F_2(P,Q,R)$

ⓐ $F_1 = \Sigma(0,1,4,7)$ $F_2 = P'Q' + R$

ⓑ $F_1 = \Sigma(1,3,4,7)$ $F_2 = PQ + QR + PR$

ⓒ $F_1 = \Pi(0,1,2,4)$ $F_2 = PQ + PR + QR$

ⓓ $F_1 = \Sigma(4,5,6)$ $F_2 = \Pi(4,5,6)$

⇒ ⓓ option is wrong definetly as ~~the~~ Σ and Π cannot be equivalent as both functions should contain same number of minterms at same places.

For ⓐ the minterms associated with F_2 will be

$\therefore F_2 = P'Q' + R$

$P'QR$	PQR
00ϕ	ϕϕ1
⇒ 001 ⇒ 1	100 ⇐
000 ⇒ 0	3 ⇐ 011
	5 ⇐ ϕ01
	7 ⇐ 111

$\therefore F_2 = \Sigma(0,1,3,5,7)$

$\therefore F_1$ and F_2 are not ~~same~~ dual $\therefore$ it is incorrect

For ⓑ the minterms associated with F_2 will be

$\therefore F_2 = PQ + QR + PR$

PQR	PQR	PQR
11ϕ	ϕ11	1ϕ1
⇒ 110 = 6	⇒ 011 = 3	⇒ 101 = 5
111 = 7	⇒ 111 = 7	⇒ 111 = 7

$\therefore F_2 = \Sigma(6,7,3,5)$

$\therefore F_1 \neq F_2$ $\therefore$ it is incorrect

Hence, ⓒ is correct option.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.15 Solved Problems:

(Q) Minterm expression of $F(P,Q,R) = PQ + Q\bar{R} + P\bar{R}$ is

(a) $m_2 + m_4 + m_6 + m_7$

(b) $m_0 + m_1 + m_6 + m_7$

(c) $m_0 + m_1 + m_3 + m_5$

(d) $m_2 + m_3 + m_4 + m_5$

⇒ On extracting minterms from SOP

$F(P,Q,R) = PQ + Q\bar{R} + P\bar{R}$

PQ → PQR: 11 ϕ ⇒ 110 = 6, 111 = 7

$Q\bar{R}$ → PQR': ϕ 1 0 ⇒ 010 = 2, ⇒ 110 = 6

$P\bar{R}$ → PQR': 1 ϕ 0 ⇒ 110 = 6, ⇒ 100 = 4

∴ minterms are $\Sigma(2,4,6,7)$

Hence, (a) is correct //

(Q) Let $F(w,x,y,z) = \Sigma(0,4,5,7,8,9,13,15)$

Which of the following expression is NOT equal to F.

(a) $x'y'z' + w'xy' + wyz + xz$

(b) $w'y'z' + wx'y' + xz$

(c) $w'y'z' + wx'y' + xyz + xy'z$

(d) $x'y'z' + wx'y' + w'y$

(i) (a) only (ii) (b) and (d) (iii) (c) and (d) (iv) (d) only

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

⇒ Lets check the minterms for each term in the expression.

For xz ⇒

w	x	y	z	
ϕ	1	ϕ	1	
0	1	0	1	= 5
0	1	1	1	= 7
1	1	0	1	= 13
1	1	1	1	= 15

∴ all of them are in function F
∴ the term xz will be these in exp.

For $w'y$ ⇒

w'	x	y	z	
0	ϕ	1	ϕ	
0	0	1	0	= 2
0	0	1	1	= 3
0	1	1	0	= 6
0	1	1	1	= 7

→ these are not in F
∴ (d) should be there in answer
∴ we can eliminate (i)

Similiarly on checking for (b) and (c) we can find that they cannot be stayed with (d) as they are valid

∴ (iv) is the correct option.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(a) What are the maxterms for given function

$F(P, Q, R) = (P+Q)(\bar{P}+\bar{R})$

⇒ For $(P+Q)$ ⇒

P	Q	R	
0	0	ϕ	
0	0	0	= 0
0	0	1	= 1

For $(\bar{P}+R)$ ⇒

P	Q	R	
1	ϕ	1	
1	0	1	= 5
1	1	1	= 7

∴ maxterms are, $\Pi(0, 1, 5, 7)$

While, minterms are, $\Sigma(2, 3, 4, 6)$ //

(a) Extract maxterms for SOP expression given as

$F(P, Q, R) = P + \bar{Q}\bar{R}$

⇒ For term P :

P	Q	R	
1	ϕ	ϕ	
1	0	0	= 4
1	0	1	= 5
1	1	0	= 6
1	1	1	= 7

For term $\bar{Q}\bar{R}$:

P	Q	R	
ϕ	0	0	
1	0	0	= 4
0	0	0	= 0

∴ minterms are: $\Sigma(0, 4, 5, 6, 7)$

maxterms are ⇒ $\Pi(1, 2, 3)$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

3.16 Gate Questions on Boolean Expressions:

(Q) 2011 $(P+\bar{Q}+\bar{R})(P+\bar{Q}+R)(P+Q+\bar{R})$

(a) $\bar{P}Q+\bar{R}$

(b) $P+\bar{Q}\bar{R}$

(c) $\bar{P}Q+R$

(d) $PQ+\bar{R}$

⇒ The canonical POS is given to us, on extracting maxterms we will get

$P+\bar{Q}+\bar{R} \Rightarrow 011 = 3$

$P+\bar{Q}+R \Rightarrow 010 = 2$ } maxterms

$P+Q+\bar{R} \Rightarrow 001 = 1$ }

$F(P,Q,R) = \Pi(1,2,3)$

$= \Sigma(0,4,5,6,7)$

Now, check for options

For term R:

P	Q	R	
ϕ	ϕ	1	
0	0	1	= 1
0	1	1	= 3
1	0	1	= 5
1	1	1	= 7

∴ it includes 1 and 3 ∴ R will not be in the solution ∴ (c) is wrong

For term $\bar{P}Q$:

P	Q	R	
0	1	ϕ	
0	1	0	= 2 } this will also not include
0	1	1	= 3 }

∴ (a) can be removed.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

For term $\bar{R}$:

P	Q	R	
ϕ	ϕ	0	
0	0	0	= 0
0	1	0	= 2
1	0	0	= 4
1	1	0	= 6

$\therefore$ 2 is maxterm

$\therefore$ $\bar{R}$ will not be there

$\therefore$ Correct option is (b)

2015

(8) $F = P' + QR$

S_1 : $F = \Sigma(4,5,6)$

S_2 : $F = \Sigma(0,1,2,3,7)$

S_3 : $F = \Pi(4,5,6)$

S_4 : $F = \Pi(0,1,2,3,7)$

	S_1	S_2	S_3	S_4
(a)	F	T	T	F
(b)	T	F	F	T
(c)	F	F	T	T
(d)	T	T	F	F

$\Rightarrow$ For term $\bar{P}$:

P	Q	R	
0	ϕ	ϕ	
= 0	0	0	= 0
0	0	1	= 1
0	1	0	= 2
1	1	1	= 3

For term $\bar{QR}$:

P	Q	R	
ϕ	1	1	
= 0	1	1	= 3
= 1	1	1	= 7

$\therefore$ $\Sigma(0,1,2,3,7)$ and $\Pi(4,5,6)$

gate.appliedcourse.com
gatecse@appliedcourse.com

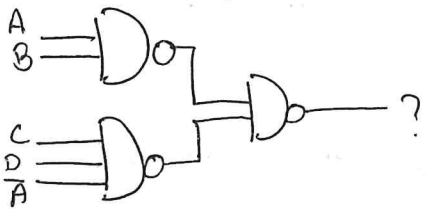
APPLIED COURSE
Ph: +91 844-844-0102

∴ only S_2 and S_3 are true

∴ Correct option is (c) //

3.18 Demorgan law and Gate problems:

(Q)

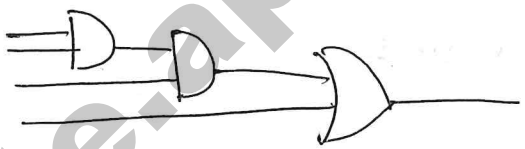


⇒ ∵ it is NAND - NAND

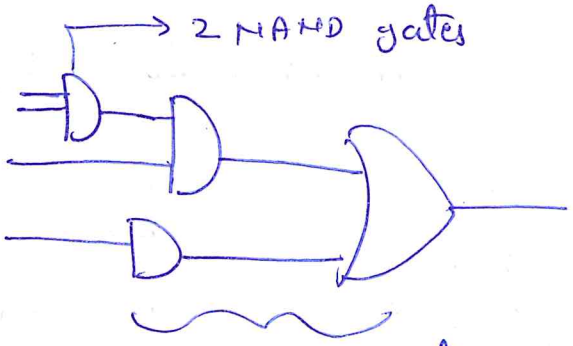
∴ it is simply AND-OR realization

∴ Eqn will be $AB + CD\bar{A}$.

(Q) Minimum no. of NAND gates (2-i/p) are required for following circuit diagram.



⇒



∴ 3+2 = 5 NAND gates are required //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

IV LOGIC GATES

4.1 Logic system Introduction:

- Positive logic system (Positive most voltage – logic 1) } Normal computation
- Negative logic system (Negative most voltage – logic 0) } RS 232 serial computation and communication

Ex:- 0V → logic 0, 5V → logic 1

Ex:- → 5V → logic 0, 10V → logic 1 } +ve most voltage will be logic 1 //

Ex:- → −12V → logic 0, +12V → logic 1

Whenever any one of the i/p is high, o/p is high

2-i/p OR gate

A	B	Y
(L) 0	0 (L)	L
(L) 0	1 (H)	H
(H) 1	0 (L)	H
(H) 1	1 (H)	H

+ve logic

A	B	Y
0	0	0
0	1	1
1	0	1
1	1	1

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

2-i/p OR gate in -ve logic system:

A	B	Y
L	L	L
L	H	H
H	L	H
H	H	H

-ve logic →

A	B	Y
1	1	1
1	0	0
0	1	0
0	0	0

AND gate

Hence, +ve logic AND or +ve logic OR will behave as -ve logic OR or -ve logic AND respectively.

4.2 Logic Gates: AND, OR and NOT:

Basic Gates: NOT, AND, OR

Real world problem
↓
Boolean exp (truth table)
↓
H/w realization (gates)
↓
Solution

* NOT Gate: O/p is complement of i/p

diagram:

Truth Table:

A	NOT
0	1
1	0

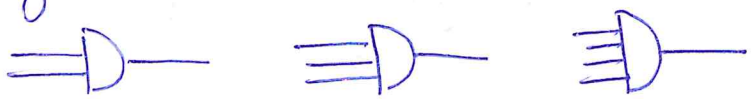
$f(A) = \bar{A}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* AND gate: Whenever all i/p's are '1', o/p is '1'

Circuit diagram:



Truth table:

A	B	AND
0	0	0
0	1	0
1	0	0
1	1	1

Expression:

$f(A,B) = AB \longrightarrow SOP = \Sigma(3)$

$= (A+B)(A+B')(A'+B) \longrightarrow POS = \Pi(0,1,2)$

- AND gate has only one minterm.
- AND gate is use to isolate the bit.

ex- R is an 8-bit register ($R_7\ R_6\ R_5\ R_4\ R_3\ R_2\ R_1\ R_0$)

We need to check whether 3 bit of R is '1'?

	R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0
AND mask:	0	0	0	0	0	1	0	0
	0	0	0	0	0	R_2	0	0 ← result

Result will be 0 only when all 8 bits are zero
result will be non-zero when R_2 bit is 1.

R → 32 bit register, whether most significant bit is 1 or 0

R_{31} → No. is +ve (if R_{31} is 0)
No is -ve (if R_{31} is 1)

Mask: 1000... 0
31 30 ... 0

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

R AND Mask ⇒ Result is Zero ⇒ R is a +ve Num
⇒ Result is Non-zero ⇒ R is a -ve Num.

* OR Gate: If any one of i/p is '1' the o/p is '1'.

Circuit diagram:

Truth Table:

A	B	OR
0	0	0
0	1	1
1	0	1
1	1	1

Expression:

$f(A,B) = A + B \longrightarrow POS \Rightarrow \pi(0)$

$= A'B + AB' + AB \rightarrow SOP = \Sigma(1,2,3)$

(8) Which of the following is equal to ⊗ on

$P'Q + PQ' + PQ = ?$

(a) $P + Q'$
(b) $P' + Q$
(c) $P + Q$
(d) PQ

⇒ The given exp is SOP of OR gate, its equivalent will be POS of OR gate ie $P+Q$ (c)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

→ OR gate is use to mask the bits

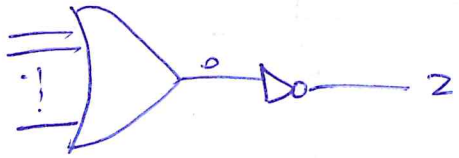
Ex- IP address given to us is : 192.0.6.100

Subnet mask : 255.255.255.0

IP and subnet mask : 192.0.6.0

R_1 operation R_2 whether result is zero or not

$R_3 = R_1$ op R_2



Zero flag

ie if Zero flag = 1 ⇒ Result is zero

if Zero flag = 0 ⇒ Result is non-zero.

4.3 Universal Gates: NAND, NOR:

- → NAND gate : Negation of AND
- → NOR gate : Negation of OR

→ All basic gates are realizable with universal gate

→ Any boolean function is realized with universal gates (multi-level) only.

→ Sum of product expression is realized with NAND-NAND logic

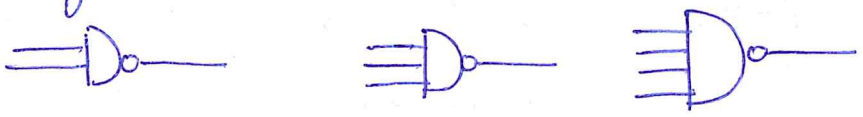
→ Product of sum is realized with NOR-NOR logic

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* **NAND gate:** When any one of i/p is zero, o/p is high (1)

Circuit diagram:



Truth table:

A	B	NAND
0	0	1
0	1	1
1	0	1
1	1	0

Expression:

$\overline{AB} = (AB)' = (A \cap B)' = A \uparrow B$

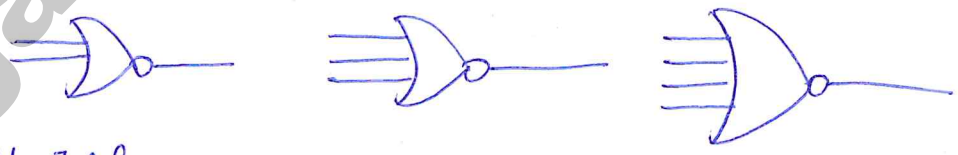
$F(A,B) = A'B' + A'B + AB'$

$= (A' + B')$

i.e. $\boxed{0\phi = \phi 0 = 1}$

* **NOR gate:** When any one of i/p is 1, o/p will be 0.

Circuit diagram:



Truth Table:

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Expressions:

$\overline{A+B} = (A+B)' = (A \vee B)' = A \downarrow B$

$F(A,B) = A'B'$ — SOP

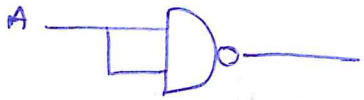
$= (A+B')(A'+B)(A'+B')$ → POS

$\Sigma(0) = \Pi(1,2,3)$

4.4 NAND-NAND, NOR-NOR Realizations:

* **NAND-NAND Realization:**

(i) NOT Gate -

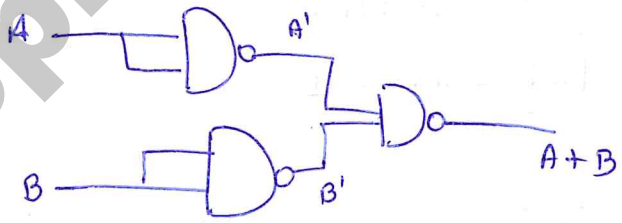


$\Rightarrow \overline{A \cdot B} = A' + B'$

$\overline{A \cdot A} = A' + A' = A'$

(ii) OR Gate -

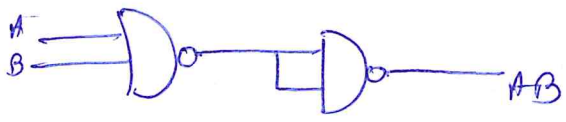
$(A+B) = (A' \cdot B')'$



Hence, in order to realize OR gate we need 3 NAND gates.

(iii) AND Gate -

$AB = (\overline{AB})'$



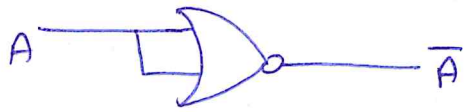
Hence, in order to realize AND gate we need 3 NOR gates.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* NOR-NOR Realization:

(i) NOT gate:

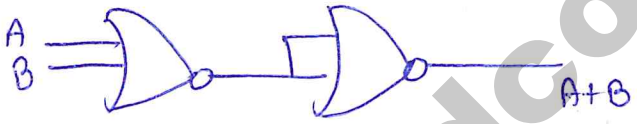


$(A+B)' = A'B'$

$(A+A)' = A' \cdot A' = A'$

Hence, 1 NOR gate is sufficient to realize NOT gate

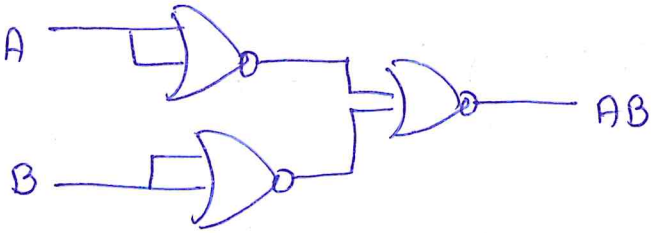
(ii) OR gate:



$(A+B) = (\overline{A+B})'$

Hence, 2 NOR gates are required to realize OR gate

(iii) AND gate:



$AB = (A'+B')'$

Hence, 3 NOR gates are required to realize AND gate

gate.appliedcourse.com
gatecse@appliedcourse.com

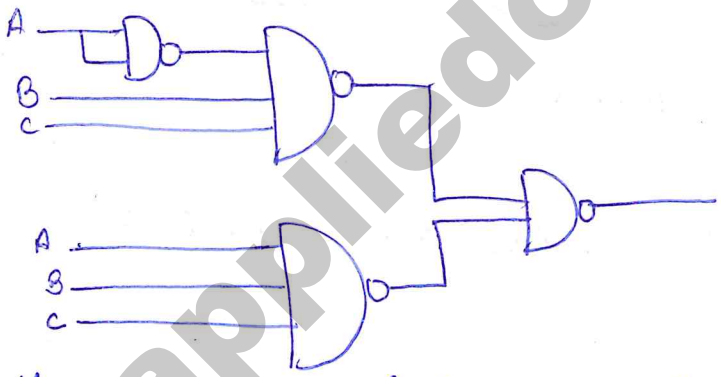
APPLIED COURSE
Ph: +91 844-844-0102

Universal Gate	Number of universal gates needed for basic Gate		
	Basic Gate		
	AND	NOT	OR
NAND	2	1	3
NOR	3	1	2

$\rightarrow (A'+B')'$

(i) No. of NAND gates required for the given function
$F(A, B, C) = A'BC + ABC$

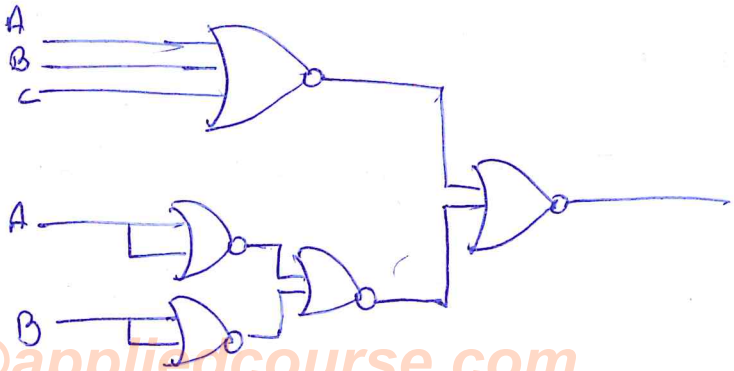
$\Rightarrow$



Hence, 4 NAND gates are required //

(ii) $F(A,B,C) = (A+B+C)(A'+B')$. What is the no. of NOR gates required?

$\rightarrow$



hence, 5 NOR gates are required //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

4.5 Other gates ExOR and Ex-NOR:

* **Ex-OR gate:** whenever odd no. of i/ps are high (1), then o/p is high (1)

Circuit diagram

Expression -

$A \oplus B = A'B + AB'$

↳ Ex-OR symbol

Truth table:

A	B	Ex-OR
0	0	0
0	1	1
1	0	1
1	1	0

$F(A,B) = A'B + AB' = \Sigma(1,2)$

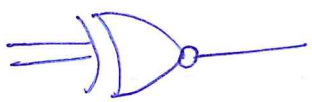
$= (A+B)(A'+B') = \Pi(0,3)$

↓

$A \oplus B$

* **Ex-NOR gate:** (Negation of Ex-OR)
whenever odd no. of i/p are high(1), then o/p is low(0)

Circuit diagram:



Truth Table:

A	B	Ex-NOR
0	0	1
0	1	0
1	0	0
1	1	1

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Expression:

$$f(A,B) = A'B' + AB = \Sigma(0,3)$$
$$= (A+B')(A'+B) = \pi(2,1)$$

$A \odot B$ → Symbol of Ex-NOR

NOTE -

* For odd number of inputs Ex-OR and Ex-NOR have equality relation.
* For even number of inputs Ex-OR and Ex-NOR have complementation relation.

Ex - for 3 i/p system A, B and C,

$$A \oplus B \oplus C = A \odot B \odot C$$

for 2-i/p system A, B,

$$A \oplus B = (A \odot B)'$$

or, $$A \odot B = (A \oplus B)'$$

Properties of Ex-OR:-

* Consider 2 binary images of size 2×2

0	1
1	0

Image A (2×2)

0	1
1	0

Image B (2×2)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

If we need to check, whether the two images A and B are same or not, then we can check for these Ex-OR value.

If $A \oplus B = 0 \Rightarrow$ Pattern of A matches with pattern B

Hence, it is used for pattern matching.

* EXOR operation is used for data Encryption.

* If both i/p's are complemented then, Ex-OR remains same.

* If one i/p is complemented then, Ex-OR become Ex-NOR.

* If 0 is Ex-ORed with P (then, it remains same) $\Rightarrow 0 \oplus P = P \oplus 0 = P$

* If 1 is Ex-ORed with P (then, it gets complemented) $\Rightarrow 1 \oplus P = P \oplus 1 = P'$

* $x \oplus y = x' \oplus y = x' \odot y = x \odot y'$

* $x \oplus x = 0$; $x \oplus x' = 1$; $x \oplus 0 = 0 \oplus x = x$; $x \oplus 1 = 1 \oplus x = x'$

Imp
* $x \oplus y \oplus xy = x + y$

$\downarrow$

$x + y + xy$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

* Ex-OR follows commutative property.
  ie $x \oplus y = y \oplus x$

* if $x \oplus y = z$, then
  $z \oplus x = y$; $z \oplus y = x$

* $x \oplus y \oplus z \oplus xy \oplus yz \oplus zx \oplus xyz = x + y + z$

Properties of Ex-NOR:

* It is dual of Ex-OR
* $x \odot y = x' \odot y' = x \oplus y' = x' \oplus y$
* $x \odot x = 1$; $x \odot x' = 0$
* $x \odot 1 = 1 \odot x = x$; $x \odot 0 = 0 \odot x = x'$
* $\underbrace{x \odot y \odot (x+y)}_{\downarrow} = xy$
  $x \cdot y \cdot (x+y)$
* $x \odot y = y \odot x$ (Ex-NOR follows commutative)

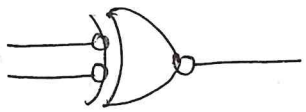
* if $x \odot y = z$, then
  $z \odot x = y$; $z \odot y = x$

* $x \odot y \odot z \odot (xy) \odot (yz) \odot (zx) \odot (x+y+z) = xyz$

gate.appliedcourse.com
gatecse@appliedcourse.com

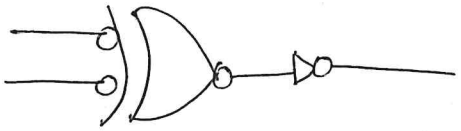
APPLIED COURSE
Ph: +91 844-844-0102

(8) Which of the following is not equal to 2 i/p Ex-NOR?

(a) 

(b)

(c)

(d) 

$\Rightarrow$

For (a) $\because x' \odot y' = x \odot y$

$\therefore$ it is Ex-NOR only

For (b) $\because x' \odot y = x \odot y$

$\therefore$ it is also Ex-NOR

For (c) $\because x \odot y' = x \odot y$

$\therefore$ it is also Ex-NOR

For (d) $(x' \odot y')' = (x \odot y)' = x \oplus y$

(Ex-NOR − NOT = Ex-OR)

$\therefore$ it is not Ex-NOR, Hence it is Correct //

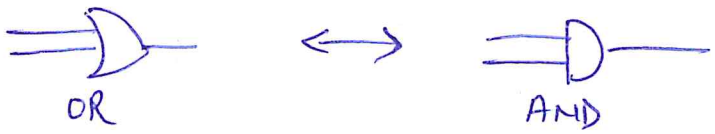
gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

4.7 Interconversion of logic systems:

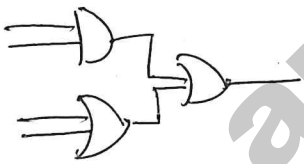
Positive logic 'OR' will behave as negative logic 'AND' vice versa.



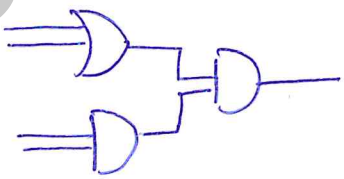
(Inversion remains same)

While, NOT remains same

(Q) Convert the following realization from +ve logic system to -ve logic system.



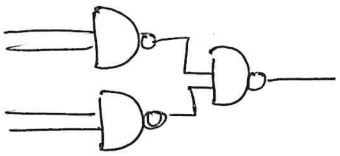
⇒ The AND is replaced with OR and vice versa.



gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) Convert the following realization from +ve logic system to -ve logic system

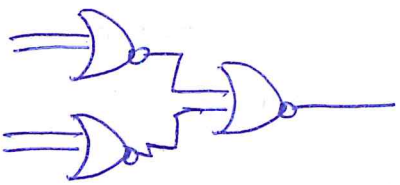


⇒ ∵ NAND ⇒ AND + NOT

↓ ↓

OR + NOT ⇒ NOR

∴ its equivalent will be



(These 2 are dual of each other)

Note:

Duality characteristics is obtained by interchanging logic system realization.

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

V K-Maps (Karnaugh Maps)

5.1 K-Maps Basics:

It provides the systematic way of minimization of boolean expression. It is practical / pictorial approach for minimization of boolean expressions

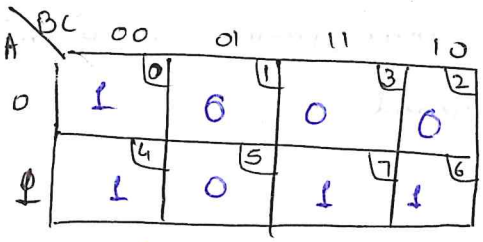
* Completely specified functions: A boolean function is said to be completely specified if its precisely define the o/p for given any i/p combination.
* Incompletely specified functions: For certain i/p combinations we cannot tell whether the o/p will be 1 or 0, such a case is called "Don't care". Thus when don't cares are present in the K-Map then those functions are called incompletely specified function.

* Minimization of minterms is called as minimization of SOP expression while minimization of maxterms is called as minimization of POS expression.

gate.appliedcourse.com
gatecse@appliedcourse.com

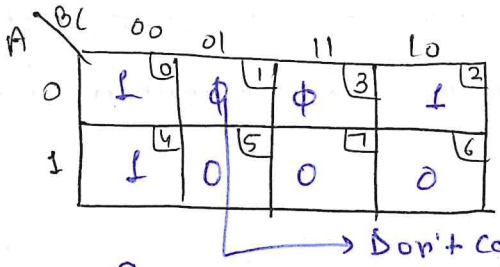
APPLIED COURSE
Ph: +91 844-844-0102

Example of completely and incompletely specified functions:



Completely specified function

$F(A,B,C) = \Sigma(0,4,6,7)$



Don't cares

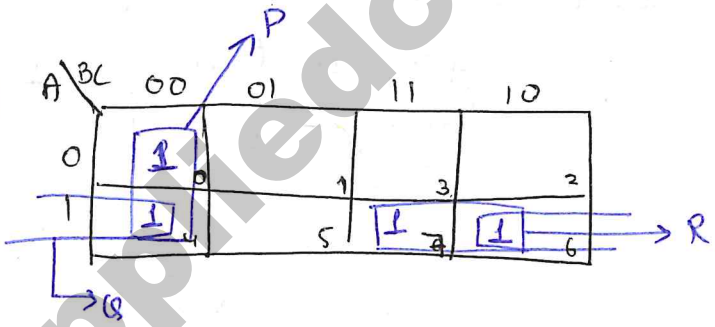
Incompletely specified function

$F(A,B,C) = \Sigma(0,2,4) + \Sigma_\phi(1,3) = \Pi(5,6,7) + \Sigma_\phi(1,3)$

Don't cares

→ SOP minimization:

ex - $F(A,B,C) = \Sigma(0,4,6,7)$



		minterms covered
Prime Implicant →	P =	{0, 4}
	Q =	{4, 6}
	R =	{6, 7}

⇒ Prime implicants are formed adjacent cells of K-maps (which are at unit distance) containing minterms (adjacent cells means that they differ by only 1 bit)

⇒ A prime implicant is said to be <u>essential</u> if it covers atleast one minterm which is not being covered by any other prime implicant.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

In the above K-map, the minterm 0 and 7 are covered by only P and R respectively. Thus prime implicant P and R are essential prime implicant.

⇒ The prime implicant which is not essential is called redundant prime implicant. In the given example, the prime implicant Q is the redundant prime implicant.

NOTE:

Minimal cover = Minimal Expression
= All essential prime implicants + Prime implicants which are needed to cover remaining minterms, if any.

it is responsible for telling that how many minimal expression are possible.

For the above example, we are not in need to add the remaining Prime implicants as all the minterms are already being covered in Essential prime implicant. Thus only one minimal expression is possible.

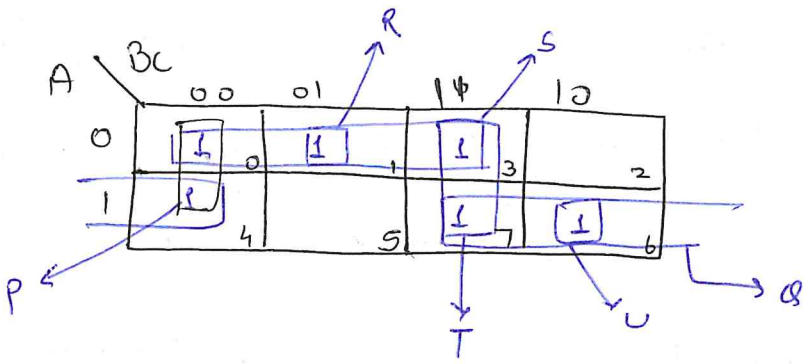
minimal expression = P + R = $\bar{B}\bar{C} + AB$
P = {0, 4}, R = {6, 7}
2 literals, 2 literals
∴ total 4 literals

$f(A,B,C) = \Sigma(0,4,6,7)$

gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Example,



Here, we have 6 prime implicants and none of them are essential prime implicants (~~no prime~~ as every minterm is covered by more than one prime implicant). Such prime implicants are called as cyclic prime implicants.

NOTE- What is implicant?

Implicant is the one which result, when one literal is removed from prime implicant.

In the above example, the implicant of 'P' is $\bar{B}\bar{C}$ (here A is not the part of it they, it is an prime implicant)

5.2 K-maps Part-I:

→ K-maps are pictorial description of truth tables

→ A truth table with n variables contain i/p's 2^n rows and one output column.

→ K-map of n-variables contains 2^n cells (denote i/p) and addressed using Grey code. Content of cell denote o/p.

→ The K-map is capable of minimising SOP, POS. Minimizing reduces the no. of terms and no. of literals in SOP(POS)

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

→ This in turn reduces the amount of hardware.

→ It provides systematic process of minimization.

NOTE-

Term 'literal' in terms $SOP = A'B' + AB$ are 4. The 2 terms are these and each term with 2-literals The minimized SOP = A', thus resulting in only one term and one literal.

* Minimization Procedure:

Step 1:

→ Construct prime-implicants. A prime implicant (PI) is formed by those adjacent cells of k-map that contains minterms (SOP) or maxterms (POS)

→ Two adjacent minterms form pair and eliminate 1 literal, 4 adjacent minterms form quad and eliminate 2 literals, 8 adjacent minterms form octal and eliminate 3 literals. A single term will have no literal eliminated

Step 2:

→ Identify Essential prime implicants. Essential prime implicant (EPI) is the one that covers atleast one minterm exclusively. ie not shared by any other PI.

Step 3:

→ Construct minimal expression.,

minimal cover = minimal expression
= All Essential PI's + PI's needed to cover remaining minterms if any.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

NOTE:

→ Prime Implicant that is not having any exclusive minterm coverage is called redundant prime implicant.

→ If all prime implicants are redundant then they are termed as cyclic prime implicant.

→ Subset is not a valid prime implicant.
(Pair in quad or quad in octal is a subset)

Gray code: Each cell is addressed using gray code (unit distance cyclic and self reflective)

Unit distance: Only one bit change in adjacent bit pattern

Cyclic: last to first combination also differs by 1 bit

Self-reflective: Reflective property is used to construct an ~~cont~~ arbitary size gray code.

1 bit gray code ≡ 1-bit binary code
(0, 1) (0, 1)

1-bit
0
1

2-bit
0 0
0 1
1 1
1 0
self reflective (mirror)

3-bit
0 0 0
0 0 1
0 1 1
0 1 0
(mirror)
1 1 0
1 1 1
1 0 1
1 0 0

4-bit
0 0 0 0
0 0 0 1
0 0 1 0
0 0 1 1
0 1 1 0
0 1 1 1
0 1 0 1
0 1 0 0
(mirror)
1 1 0 0
1 1 0 1
1 1 1 1
1 1 1 0
1 0 1 1
1 0 1 0
1 0 0 1
1 0 0 0

Gray code

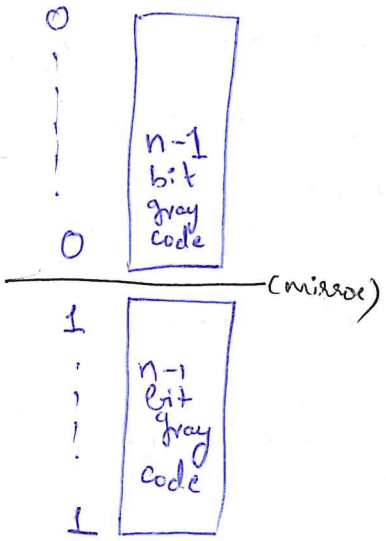
n-1 bit gray code with self reflection → n bit gray code.

2^n n-bit gray code, MSB is same as n-bit binary code
(i.e $g_{n-1} = b_{n-1}$)

gate.appliedcourse.com
gatecse@appliedcourse.com

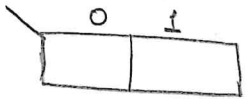
APPLIED COURSE
Ph: +91 844-844-0102

n-bit gray code



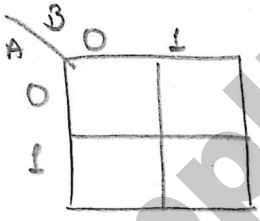
→ K-Maps -

* 1-variable :



$= 2^1$ cells $= 2$ cells

* 2-variable:



$= 2^2$ cells $= 4$ cells

* 3-variable :

A \ BC	00	01	11	10
0	000 = 0	001 = 1	011 = 3	010 = 2
1	100 = 4	101 = 5	111 = 7	110 = 6

→ These are cell addresses

$= 2^3$ cells $= 8$ cells

↳ either 2 rows and 4 colums or 4 rows and 2 Colums.

* 4-variable:

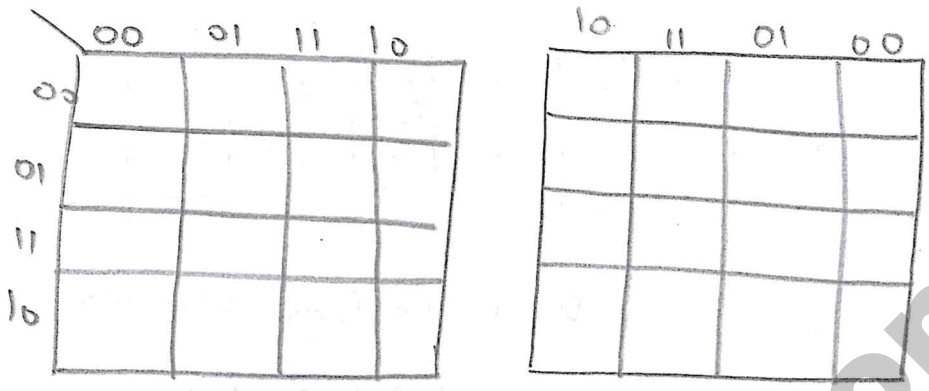
AB \ CD	00	01	11	10
00	0000 = 0	0001 = 1	0011 = 3	0010 = 2
01	0100 = 4	0101 = 5	0111 = 7	0110 = 6
11	1100 = 12	1101 = 13	1111 = 15	1110 = 14
10	1000 = 8	1001 = 9	1011 = 11	1010 = 10

$= 2^4$ cells $= 16$ cells

gate.appliedcourse.com
gatecse@appliedcourse.com

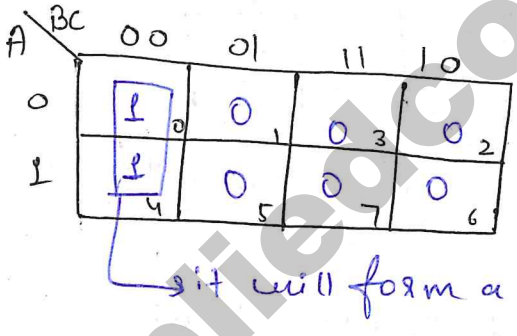
APPLIED COURSE
Ph: +91 844-844-0102

→ 5-variable :



$= 2^5$ cells $= 32$ cells.

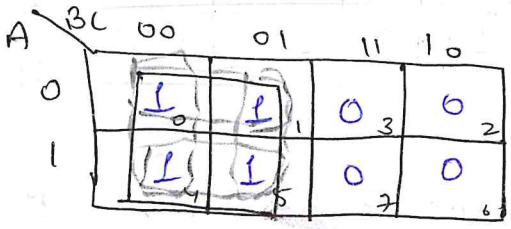
If a minterm are given as $F(A,B,C) = \Sigma(0,4)$, then the K-map will be:



The variable whose value differ in the minterm will be eliminated from the implicant. Thus A will be removed and minimal expression will be $\bar{B}\bar{C}$.

Ex- $f(A,B,C) = \Sigma(0,1,4,5)$

⇒



Here 1 quad is formed by the adjacent minterm and in that quad we can have 4 pairs. Since pairs are the subset of quad, therefore, we will not consider them.

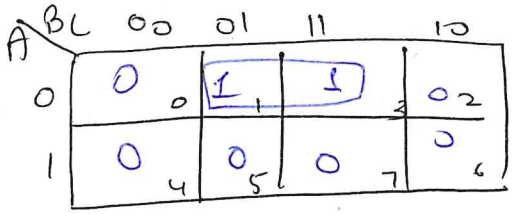
Thus minimal expression is $= \bar{B}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

ex- $f(A,B,C) = \Sigma(1,3)$

⇒

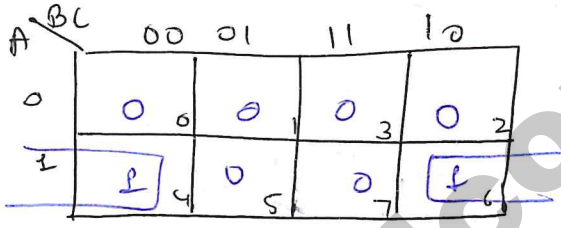


∴ B is getting changed

∴ minimal exp = $\bar{A}C$

ex- $f(A,B,C) = \Sigma(4,6)$

⇒

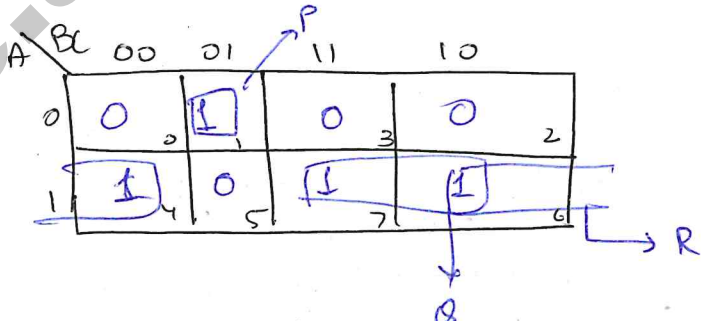


It forms a rolling pair. Here B is getting change

∴ minimal expression = $A\bar{C}$

ex = $F(A,B,C) = \Sigma(1,4,7,6)$

⇒



here we have 3 prime implicants, 1-singleton and 2 pairs.

minimal expression = $P + Q + R$

$= \bar{A}\bar{B}C + AB + A\bar{C}$

This exp. has 3 terms and 7 literals.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

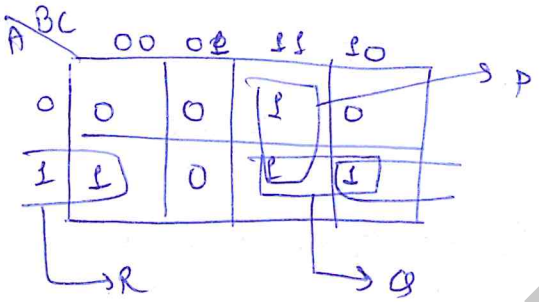
Ph: +91 844-844-0102

5.3 K-Maps Part-II :

Q) Given k-map what will be the minimal expression?

A \ BC	00	01	11	10
0			1	
1	1		1	1

$\Rightarrow$



Essential prime implicant, $\Rightarrow P = \{3, 7\} = BC$

$Q = \{6, 7\} = AB \longrightarrow$ it is redundant PI

Essential prime impl. $\Rightarrow R = \{4, 6\} = A\bar{C}$

$f(A, B, C) = \{3, 4, 6, 7\} = P + R$ + PI req. to cover remaining minterms if any

(EPI)

$= P + R$

$= BC + A\bar{C}$

$\therefore$ # terms = 2

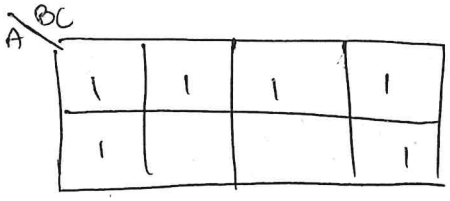
literals = 4

gate.appliedcourse.com

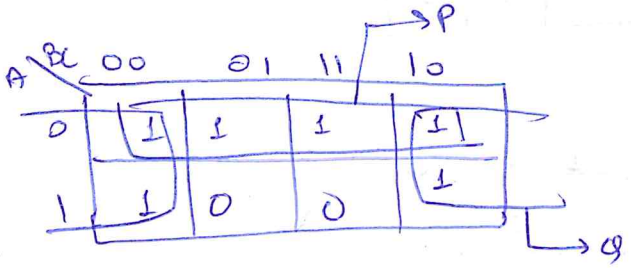
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) Minimal expression for the given K-map is?



⇒



here 2 quads are formed and both are essential

$P = \sum(0,1,2,3) = \overline{A}$

$Q = \sum(0,3,4,6) = \overline{C}$

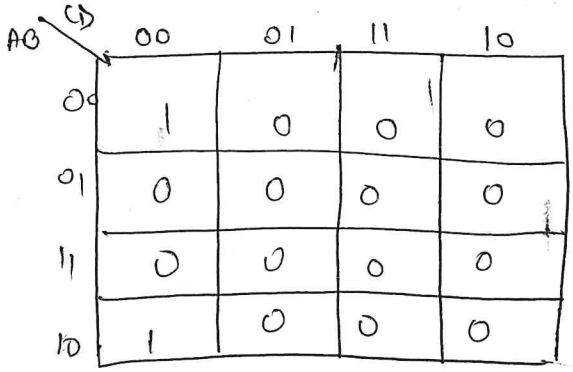
∴ minimal exp = $P + Q$

$= \overline{A} + \overline{C}$

terms = 2

literals = 2

(Q) Minimal expression for the given K-map is?



gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

⇒

AB \ CD	00	01	11	10
00	1	0	0	0
01	0	0	0	0
11	0	0	0	0
10	1	0	0	0

minimal expression will not be having variable A as its value is changing for both the terms.

∴ minimal expression = $\bar{B}\bar{C}\bar{D}$

(c&s) Minimal expression for the given K-map is?

AB \ CD	00	01	11	10
00	0 (0)	0 (1)	1 (3)	1 (2)
01	0 (4)	0 (5)	0 (7)	0 (6)
11	0 (12)	0 (13)	0 (15)	0 (14)
10	0 (8)	0 (9)	1 (11)	1 (10)

⇒

AB \ CD	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	0	0	0	0
10	0	0	1	1

→ P

One quad will be formed and since value of A and D are changing ∴ they will be eliminated

Thus minimal exp = $\bar{B}C$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) Minimal expression for the Given K-map is?

AB \ CD	00	01	11	10
00	1	0	1	0
01	0	1	0	1
11	1	0	1	0
10	0	1	0	1

⇒ Here all the terms are singleton (ie no pair, quad, octal is possible)

$$f(A,B,C,D) = \Sigma(0,3,5,6,9,10,12,15)$$

$$= A \oplus B \oplus C \oplus D$$

(Q) Minimal expression for the Given K-map is?

AB \ CD	00	01	11	10
00	1	1	1	1
01	0	0	0	0
11	0	0	0	0
10	1	1	1	1

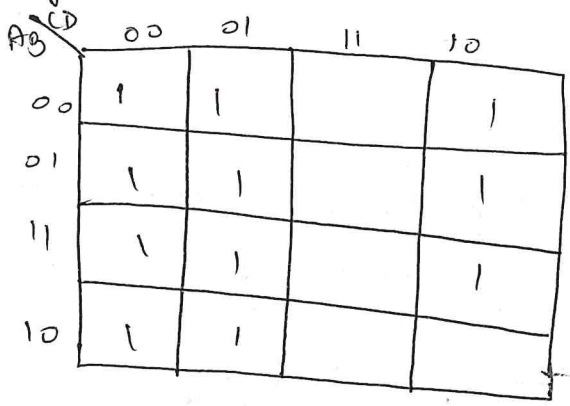
⇒ Here, one octet is possible which is rolling. Thus it will eliminate 3 variables which are A, C and D

∴ minimal expression = $\overline{B}$.

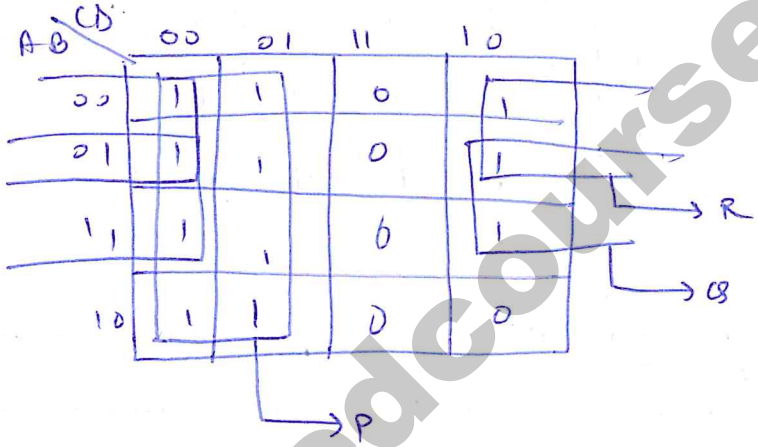
gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(b) Minimal expression for the given k-map is?



⇒

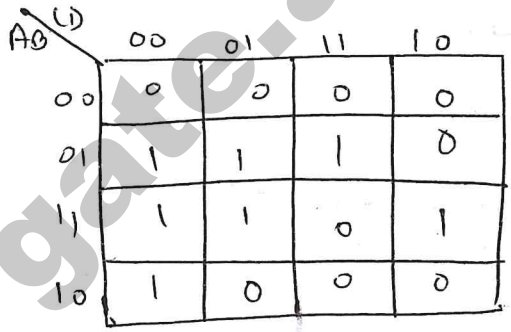


Here, 1 octet and 2 quads will be formed.

Thus minimal exp = $P + Q + R$

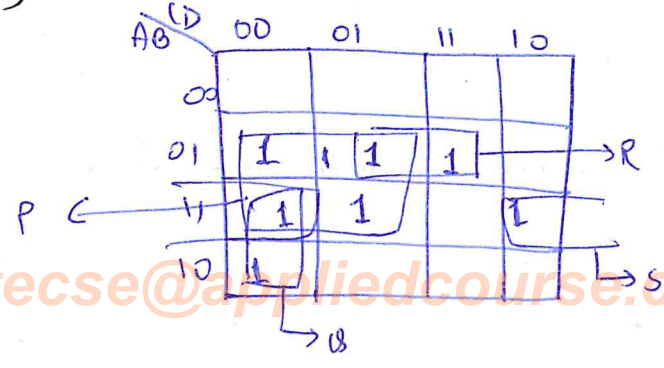
$= \bar{C} + B\bar{D} + \bar{A}\bar{D}$

(a)



minimal expression = ?

⇒



minimal exp = $P + Q + R + S$

$= B\bar{C} + A\bar{C}\bar{D} + \bar{A}BD + A\bar{B}\bar{D}$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

5.4 Previous Years Gate Questions:

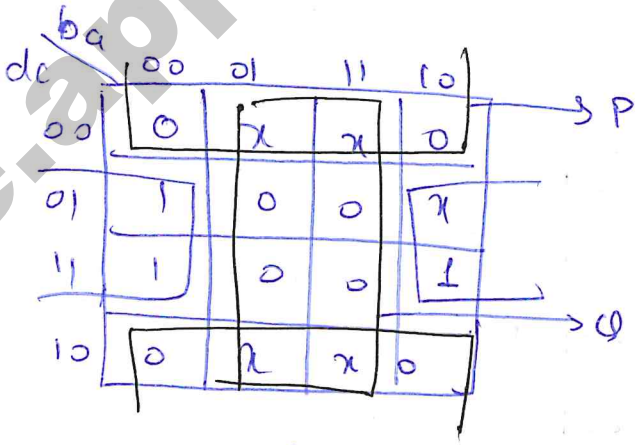
2017

(Q) Consider the K-map given below, where x represents "don't care" and blank represents 0.

dc \ ba	00	01	11	10
00		x	x	
01	1			x
11	1			1
10		x	x	

Assume for all i/p's (a, b, c, d), the respective complements (a', b', c', d') are also available. The above logic is implemented using 2-i/p NOR gates only. The minimum no. of gates required is _____

⇒



Only one quad is formed here

∴ minimal SOP exp = $C\bar{a}$

But we are asked for POS; the blanks are 0's

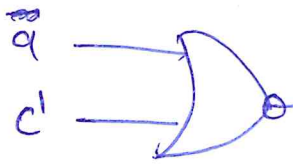
∴ 2 octets are formed

∴ minimal POS exp = $\bar{a}C\,(P \cdot Q)$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

$\therefore \bar{a}c$ could be implemented using 1 NOR gate only $\therefore$ only 1 NOR gate is required



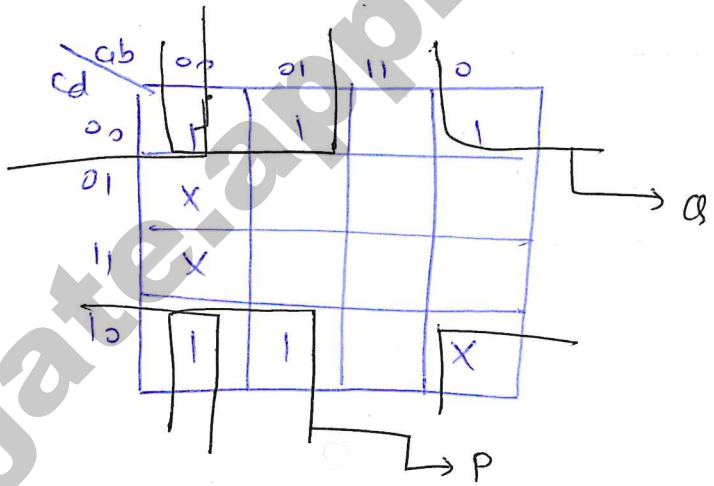
$= a' \cdot (c')' = a' \cdot c$

2008

(Q) In the K-map shown below, X denotes a don't care term. What is the minimal form of the function represented by K-map?

cd \ ab	00	01	11	10
00	1	1		1
01	X			
11	X			
10	1	1		X

$\Rightarrow$



Since 2 quads are possible

$\therefore$ minimal exp $= P + Q$

$= \bar{d}\bar{a} + \bar{b}\bar{d}$

$= \bar{d}(\bar{a}+\bar{b})$

gatecse@appliedcourse.com
gate.appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

2002

(Q) Minimum SOP expression for (w, x, y, z) shown in K-map below

yz \ wx	00	01	11	10
00	0	1	1	0
01	x	0	0	1
11	x	0	0	1
10	0	1	1	x

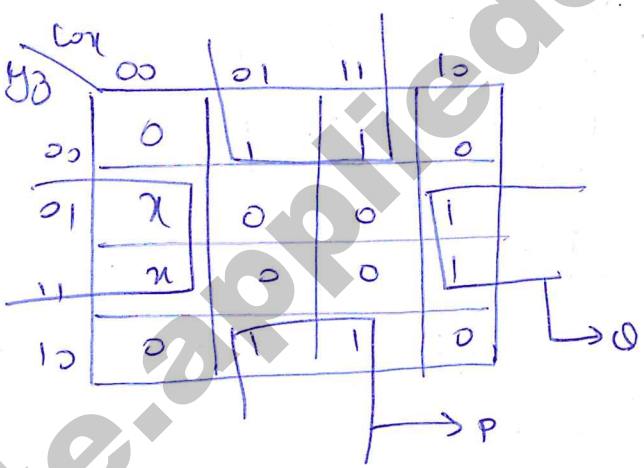
(A) $xz + y'z$

(B) $xz' + zx'$

(C) $x'y + zx'$

(D) None of the above

⇒



$\therefore P = \overline{z}x$

$Q = z\overline{x}$

$\therefore$ minimal exp $= P + Q$

$= xz' + zx'$

$\therefore$ option (B) is correct.

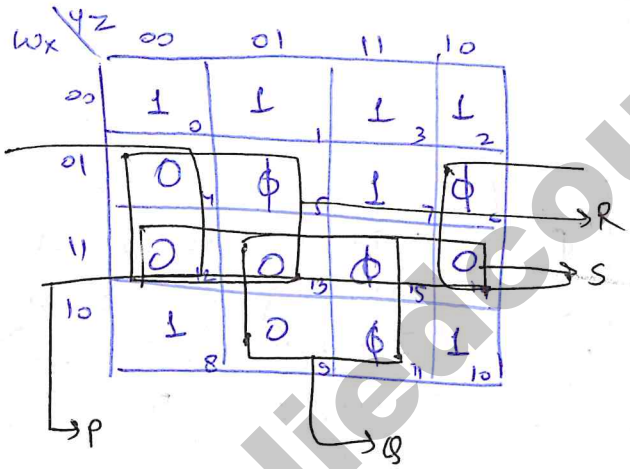
gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

2017
(Q) Given $f(w,x,y,z) = \Sigma(0,1,2,3,7,8,10) + \Sigma d(5,6,11,15)$ where d represents the 'don't care' conditions in K-maps. Which of the following is a minimum POS form of $f(w,x,y,z)$?

(A) $f = (w'+z')(x'+z)$
(B) $f = (w'+z)(x+z)$
(C) $f = (w+z)(x'+z)$
(D) $f = (w+z')(x'+z)$

$\Rightarrow$



Here, P and Q are EPI while R and S are redundant

$\therefore$ minimal exp $= P + Q$

$= (z+\bar{x})(\bar{z}+\bar{w})$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

5.5 Incompletely specified k-maps:

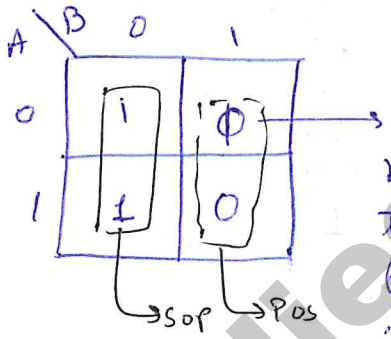
If the boolean function associate with don't cares then it is termed as incompletely specified functions.

Example, $F(w, x, y, z) = \Sigma m(0,1,2,3,7,8,10) + \Sigma d(5,6,11,15)$

Procedure:

(1) Treat only those don't case as '1' if they support minimization of SOP

Ex-



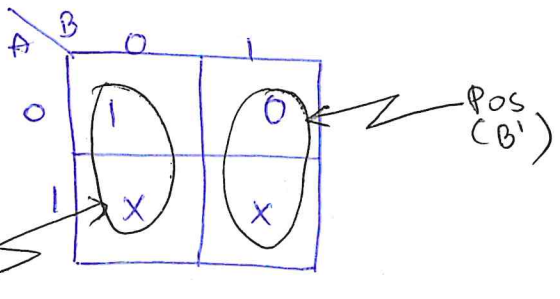
Don't care is not useful for minimizing SOP, hence, it is treated as '0'. But it is helpful in minimizing POS, as it forms pair.

(2) A don't case cannot have 1 and 0 at the same time.

$$F(A,B) = \Sigma(0,2) + \Sigma_\phi(1)$$
$$= \Pi\phi(1) + \Pi(2)$$

How to know don't care is useful for minimization of SOP or not?

⇒ Ex-



Alone minterm: only singleton is possible: $A'B'$

$SOP = \bar{B}$

$f(A,B) = \Sigma m(0) + \Sigma d(2,3) = \Pi(1) + \Pi_\phi(2,3)$

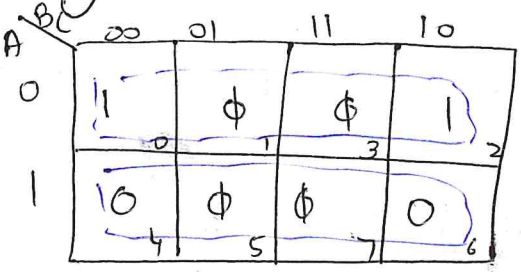
gate-appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

here, cell 2 don't care support minimization of SOP and cell 3 don't care support minimization of POS.

(Q) Which don't care cells supports minimization of SOP and POS respectively.

⇒

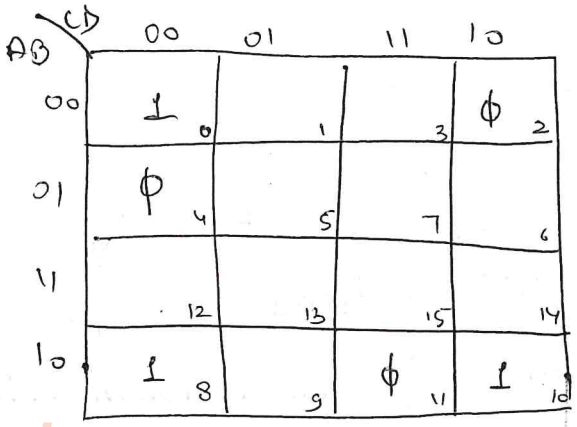


here $f(A,B,C) = \Sigma(1,2) + \Sigma_\phi(1,3,5,7)$

Don't care cell no.	Don't care value/treatment	Support minimization of SOP/POS
'1'	1	SOP
'3'	1	SOP
'5'	0	POS
'7'	0	POS

exp will be : $\bar{A}$

(Q)



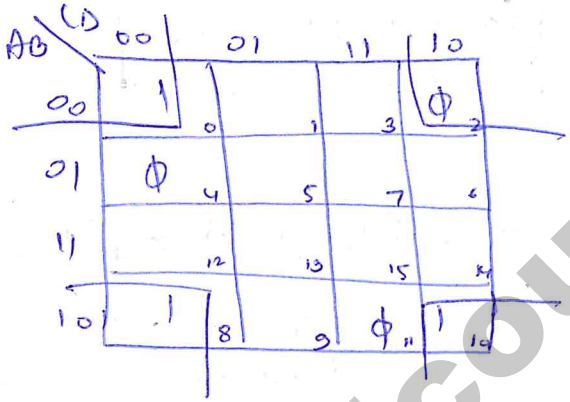
gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Which cells among 2, 4, 11 are treated as 1

(a) Only cell 2
(b) Only cell 4
(c) Only cell 11
(d) all the 2, 4, 11 cells

⇒ Here, for the given K-map we can form a quad



Thus, we only need to consider 1 don't care as 1 in order to form a quad (which is in cell 2)

∴ (a) is the correct option.

(c) How many minimal SOP expressions possible for the above K-map?

(a) 1
(b) 2
(c) 3
(d) 4

⇒ We have only 1 prime implicant and it is essential as it has covered all minterms

∴ it will have only one minimal exp

Thus, (a) is correct option.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(Q) How many prime implicants the above K-map is having? and how many of them are essential prime implicant for minimal SOP?

(a) 1, 1
(b) 1, 2
(c) 1, 3
(d) 2, 3

⇒ Since it is having only 1 PI which is also essential

∴ (a) is the correct option

The eqn will be: $f(A, B, C, D) = B'D'$

(Q) How many literals in minimal SOP of above K-map?

(a) 1
(b) 2
(c) 3
(d) 4

⇒ Since, the minimal SOP expression is: $B'D'$

∴ # literals = 2

(Q) Consider the following K-map:

AB \ CD	00	01	11	10
00	ϕ 0	1	1 3	ϕ 2
01	4	ϕ 5	ϕ 7	1 6
11	12	ϕ 13	ϕ 15	1 14
10	ϕ 8	9	1 11	ϕ 10

$f(A,B,C,D) = \Sigma(3,6,11,14) + \Sigma_\phi(0,8,5,7,13,15)$

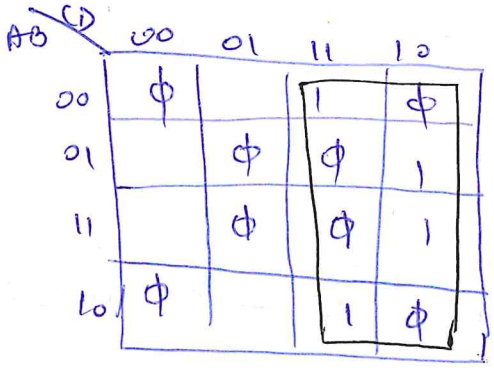
gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

(i) SOP minimization :

(Q) What are the # Prime implicants ?

⇒



Here, we can form 1 octal with the help of don't cares
∴ there is only 1 PI.
(don't care, 2, 7, 10 and 15 are treated as 1)

(Q) What are the # ~~minimal~~ Essential Prime implicants ?

⇒ Since only 1 PI is there, therefore, there ~~are~~ is only 1 essential prime implicant.

(Q) What are the # literals in minimal SOP ?

⇒ The minimal exp formed = $\bar{z}$ (as only 1 octet is there)

(Q) What are the # AND and OR gates required ?

⇒ 0, AND and OR gates are required if i/p are available in both primed and unprimed form
1, AND gate is required if primed i/p's are not available.

(Q) Minimum no. of NAND gates required ?

⇒ If i/p's are ~~are~~ not available in primed form then we need 1, 2 i/p NAND gate to realize it.

gate.appliedcourse.com
gatedata@appliedcourse.com

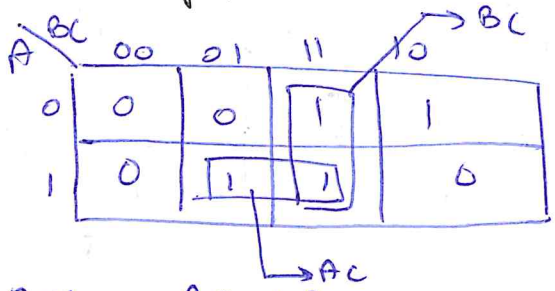
APPLIED COURSE
Ph: +91 844-844-0102

(a) Consider the K-map:

A \ BC	00	01	11	10
0	0	0	1	0
1	0	1	1	0

(I) What is the SOP exp?

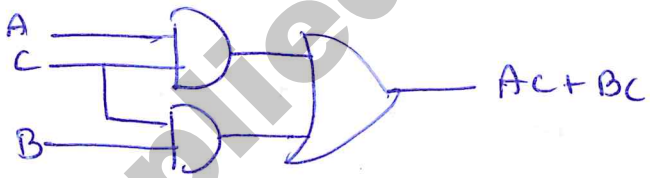
⇒



exp = $AC + BC$

(II) # AND, OR gates required?

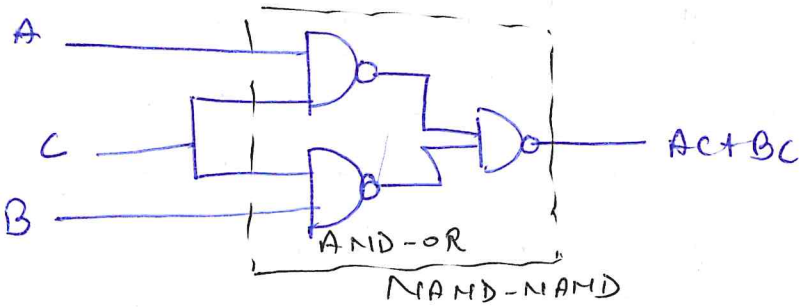
⇒ ∵ $F(A,B,C) = AC + BC$



∴ 2 AND, 1 OR gate is req.

(III) # NAND gates required?

⇒ ∴ $f(A,B,C) = AC + BC$



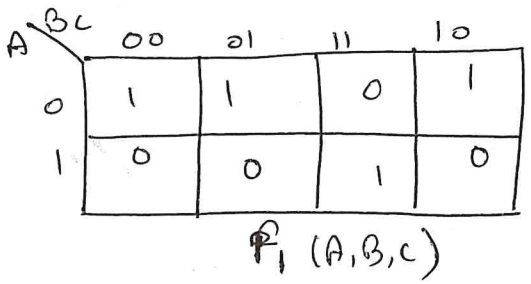
∴ 3, 2i/p NAND gates are required. //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

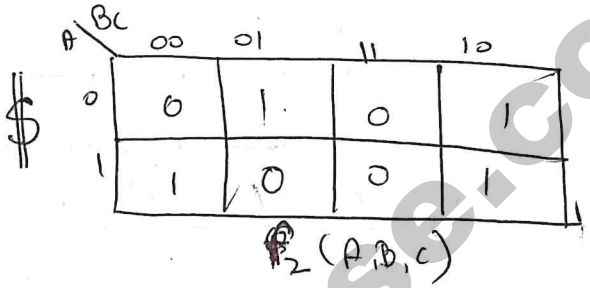
5.6 Functional K-maps:

Example: Let us consider 2 K-maps and a $ operation is applied on them.



A \ BC	00	01	11	10
0	1	1	0	1
1	0	0	1	0

$F_1(A,B,C)$



A \ BC	00	01	11	10
0	0	1	0	1
1	1	0	0	1

$F_2(A,B,C)$

If the operation $ is defined/given as:

$P \$ Q = P' + Q \Rightarrow P$ implies Q

We need to find $f_1 \$ f_2$

$\Rightarrow \because P \$ Q = 0$ iff $P = 1$ and $Q = 0$

else, 1

$\therefore$ now we can make the resultant K-map by performing the operation ~~on~~ in between respective cells.

A \ BC	00	01	11	10
0	$1' \$ 0 = 1$	$1' \$ 1 = 0$	0	0
1	0	0	1	0

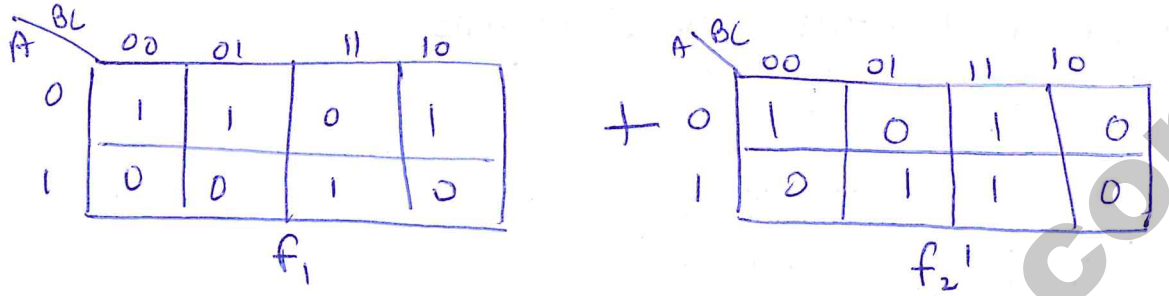
Exp will be: $A'B'C' + ABC$

gate.appliedcourse.com
gatecse@appliedcourse.com

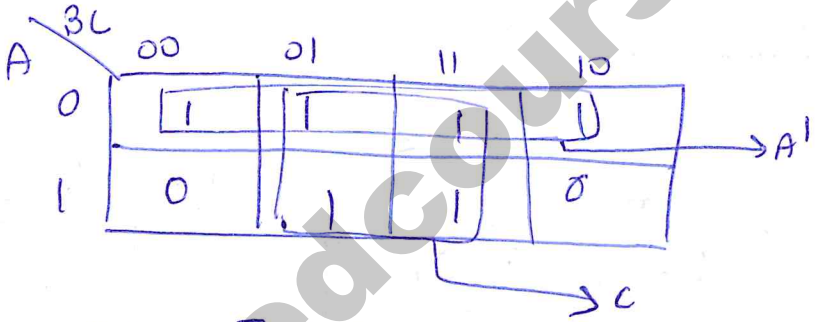
APPLIED COURSE
Ph: +91 844-844-0102

Similarly, we can perform other operations in between f_1 and f_2

ex- $f_1 + f_2' = ?$



=

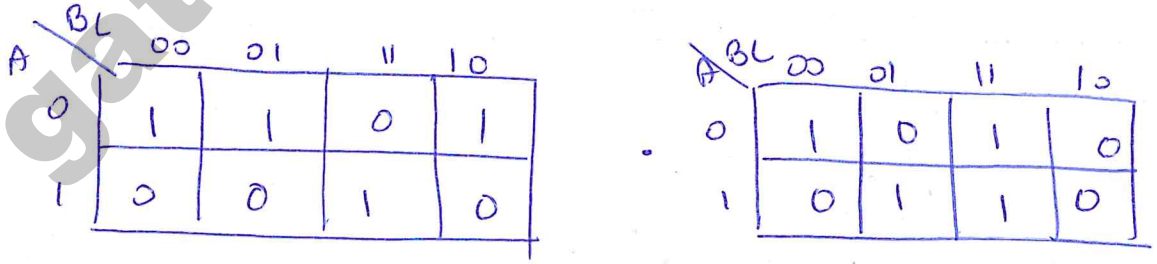


The exp will be: $\bar{A} + C$ //

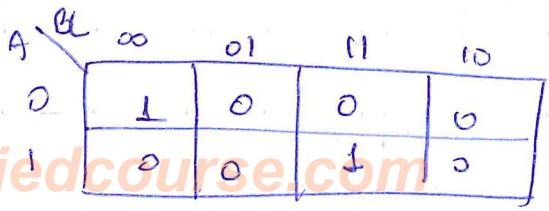
NOTE

This Concept is called functional K-map

ex- $f_1 \cdot f_2' = ?$



=



$\therefore$ Exp will be: $AB\bar{C} + ABC$ //

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

5.7 5-Variable k-map:

A 5-variable k-map is having 2^5 cells = 32 cells arranged in 4 rows and 8 columns.

3 bit

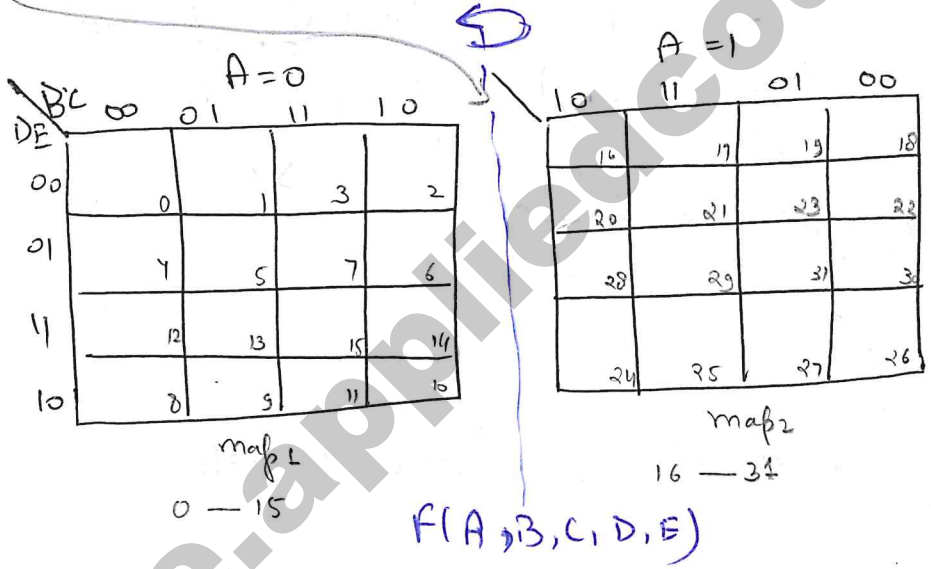
represent map 1 add

000
001
011
010

(mirror)

represent map 2 add

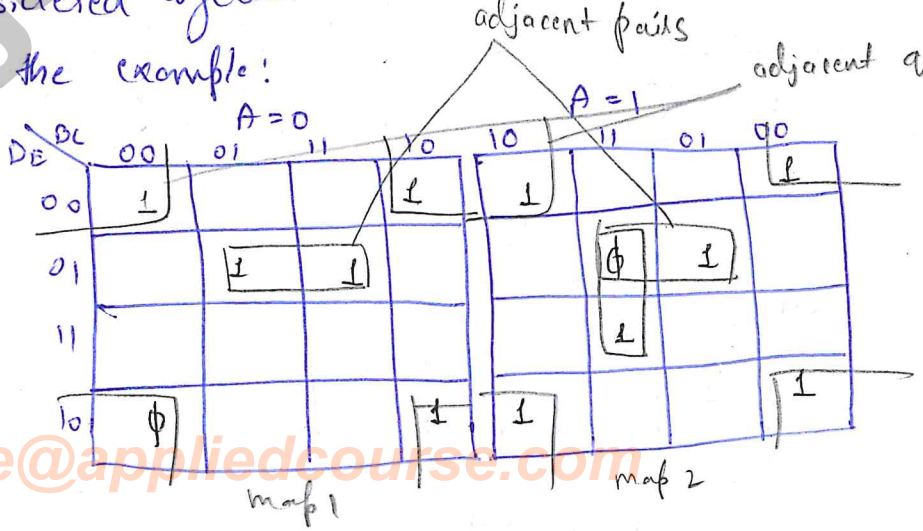
110
111
101
100



F(A, B, C, D, E)

* Each map is treated independently and then they are considered together.

Consider the example:



gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Thus, from each map we have:

map 1: 1-pair, 1-Quad (2 PI)

map 2: 2-pair, 1-Quad (3 PI)

Now if we combine 2 maps,

here, the quad of map 1 and map 2 are adjacent, therefore, they will form octet

Similarly, the adjacent pairs will form the quad.

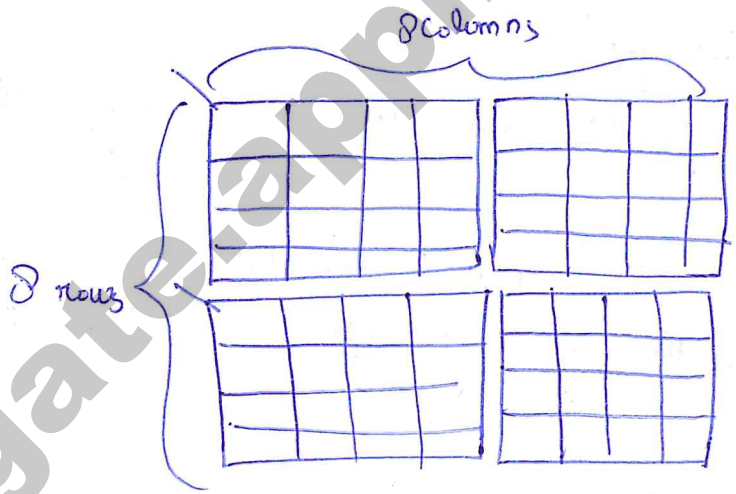
∴ Together they form: 1 → octet, 1 → quad, 1 → Pair } 3 PI

Expression = $\bar{C}\bar{E} + C\bar{D}E + ABCE$

F(A,B,C,D,E) (octet) (quad) (pair)

NOTE

* If no. of variables are 6 then it will have $2^6 = 64$ cells
  ↳ 4, 4-variable map



It includes a lot of complexity. Therefore K-maps are suitable for minimizing functions having atmost 5 variables.

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

5.8 VEM K-maps (Variable Entered Map):

Ph: +91-844-844-0102

→ VEM is capable of representing higher variable functions with lesser variable map.

→ Here cell content be; 0, 1, ϕ or any variable or function

Example:

A \ BC	00	01	11	10
0	0	1	D	D
1	0	ϕ	D'	D'

F(A,B,C,D)

,

A \ B	0	1
0	C+D	1
1	0	C'D'

F(A,B,C,D)

Minimization procedure:

Step 1: Treat all variables in the cells as '0' and get SOP.

Step 2a: Make one variable of cell as '1' with previous 1's as don't cares get SOP.

Step 2b: Multiply SOP of step 2a with concerned cell variable.

Step 3: Repeat step 2 until all cell variables are covered.

Step 4: SOP of VEM is the sum of all VEM's obtained earlier.

Example:

A \ BC	00	01	11	10
0	D	D	1	ϕ
1	0	0	D'	0

gate.appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Step 1: Set D and D' as '0'

A \ BC	00	01	11	10
0	0	0	1	ϕ
1	0	0	0	0

SOP = $\bar{A}B$

Step 2a: For D, set D = 1 and previous 1's as don't cares

A \ BC	00	01	11	10
0	1	1	ϕ	ϕ
1	0	0	0	0

SOP = ~~AB~~ $\bar{A}$

Step 2b for D: Multiply the variable with SOP

$\therefore$ SOP = DA'

Step 2a: For D', set D' = 1 and previous 1's as don't cares

A \ BC	00	01	11	10
0	0	0	ϕ	ϕ
1	0	0	1	0

SOP = BC

Step 2b for D': Multiply the variable with SOP

$\therefore$ SOP = $D'BC$

$\therefore$ SOP of VEM = SOP of step1 + SOP of step2

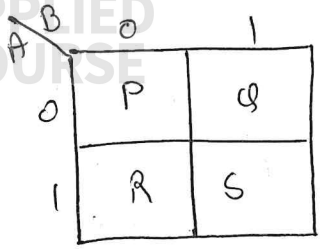
= $A'B + AD + BCD'$

gate.appliedcourse.com
gatecse@appliedcourse.com

APPLIED COURSE

Ph: +91 844-844-0102

(Q)



$F(A,B,C) = \Sigma(0,2,4,5,7) =$

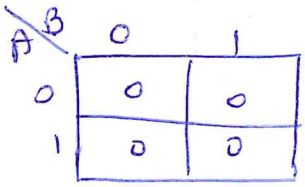
Then $P = ?$

$Q = ?$

$R = ?$

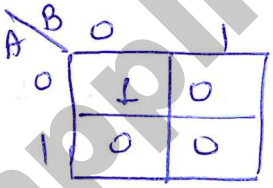
$S = ?$

$\Rightarrow$ Step 1: all variables as zero



SoP = 0

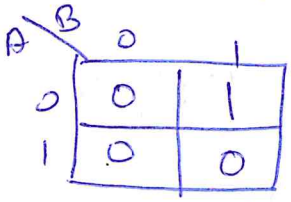
Step 2 for P: make P = 1



SoP = A'B'

final = PA'B'

Step 2 for Q: make Q = 1



SoP = A'B

final = A'BQ

gate-appliedcourse.com

gatecse@appliedcourse.com

APPLIED COURSE
Ph: +91 844-844-0102

Step 2 for R: make R = 1

A \ B	0	1
0	0	0
1	1	0

SoP = $R \cdot AB'$

Step 2 for S: make S = 1

A \ B	0	1
0	0	0
1	0	1

SoP = $S \cdot AB$

SoP of VEM = $A'B'P + A'BQ + AB'R + ABS$ —— (1)

$\therefore F(A,B,C) = \Sigma(0,2,4,5,7)$

$= \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + A\bar{B}\bar{C} + A\bar{B}C + ABC$ —— (2)
$(0) \quad (2) \quad (4) \quad (5) \quad (7)$

On equating (1) and (2) we will get:

$P = \bar{C}$ $(\bar{A}\bar{B}\bar{C} \leftrightarrow \bar{A}\bar{B}P)$

$Q = \bar{C}$ $(\bar{A}BQ \leftrightarrow \bar{A}B\bar{C})$

$R = 1$ $(A\bar{B}\bar{C} + A\bar{B}C \leftrightarrow A\bar{B}R)$

$S = C$ $(ABC \leftrightarrow ABS)$

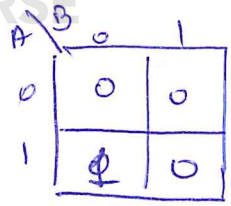
A \ B	0	1
0	C'	C'
1	1	C

SoP of VEM = ~~$A'B' + 1$~~ ?

gate.appliedcourse.com
gatecse@appliedcourse.com

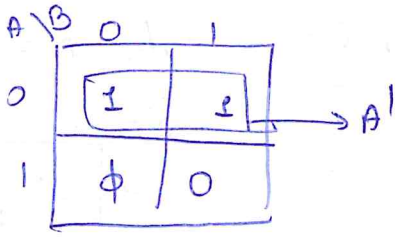
APPLIED COURSE
Ph: +91 844-844-0102

Step 1:



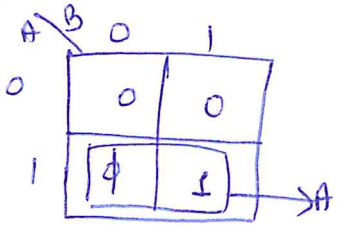
AB'

Step 2: '1's



SOP = $A'C'$

Step 3: '0's



SOP = AC

$\therefore$ SOP of VEM = $AB' + A'C' + AC$

gate.appliedcourse.com

gatecse@appliedcourse.com